Characterizing and Proving Operational Termination of Deterministic Conditional Term Rewriting Systems

Felix Schernhammer and Bernhard Gramlich

March 2009

TECHNISCHE UNIVERSITÄT WIEN

Technical Report E1852-2009-01

Theory and Logic Group, Institute of Computer Languages (E185/2)
TU Wien, Favoritenstraße 9, A-1040 Wien, Austria

Characterizing and Proving Operational Termination of Deterministic Conditional Term Rewriting Systems

Felix Schernhammer and Bernhard Gramlich*
Theory and Logic Group
Institute of Computer Languages
TU Wien, Austria

March 2009

Abstract

We investigate the practically crucial property of operational termination of deterministic conditional term rewriting systems (DCTRSs), an important declarative programming paradigm. We show that operational termination can be equivalently characterized by the newly introduced notion of context-sensitive quasi-reductivity. Based on this characterization and an unraveling transformation of DCTRSs into context-sensitive (unconditional) rewrite systems (CSRSs), context-sensitive quasi-reductivity of a DCTRS is shown to be equivalent to termination of the resulting CSRS on original terms. This result enables both proving and disproving operational termination of given DCTRSs via transformation into CSRSs. A concrete procedure for this restricted termination analysis (on original terms) is proposed and encouraging benchmarks obtained by the termination tool VMTL, that utilizes this approach, are presented. Finally, we show that the context-sensitive unraveling transformation is sound and complete for collapse-extended termination, thus solving an open problem of [Duran et al. 2008].

1 Introduction and Overview

Conditional term rewriting systems (CTRSs) are a natural extension of unconditional such systems (TRSs) allowing rules to be guarded by conditions. Conditional rules tend to be very intuitive and easy to formulate and are therefore used in several declarative programming and specification languages, such as Maude [10] or ELAN [9]. Here we focus on the particularly interesting class

*Email: {felixs,gramlich}@logic.at. F. Schernhammer has been supported by the Austrian Academy of Sciences under grant 22.361.

of *deterministic* (oriented) CTRSs (DCTRSs) which allows for extra variables in conditions and right-hand sides to some extent (corresponding to *let-constructs* or *where-clauses* in other functional-(logic) languages) and has been used for instance in proofs of termination of (well-moded) logic programs [13].

When analyzing the termination behaviour of conditional TRSs, it turns out that the proof-theoretic notion of *operational termination* is more adequate than ordinary termination in the sense that practical evaluations w.r.t. operationally terminating DCTRSs always terminate (which is indeed not true for other similar notions like *effective termination* [22]).

For the analysis of operational termination of DCTRSs usually the equivalent property of quasi-decreasingness is used [22]. In [29], [28], based on the idea of *unravelings* of [24], a transformation from DCTRSs into TRSs is proposed such that termination of the transformed TRS implies *quasi-reductivity* of the given DCTRS which in turn implies its quasi-decreasingness.

We propose an alternative definition of *quasi-reductivity* using context-sensitivity ([20]), that will be proved to be equivalent to operational termination of DCTRSs. Furthermore, we use a simple modification of Ohlebusch's transformation ([29]) that allows us to completely characterize the new property of context-sensitive quasi-reductivity of a DCTRS by means of termination of the context-sensitive (unconditional) TRS, that is obtained by the transformation, *on original terms*.

This complete characterization yields a method for disproving operational termination of DCTRSs by disproving termination of CSRSs on original terms. Moreover, we will show that the proposed transformation is sound and complete with respect to *collapse-extended* termination even if the latter notion is not restricted to original terms in the transformed system. As a corollary we obtain modularity of collapse-extended operational termination of DCTRSs.

Finally, we present an approach, which is based on the dependency pair framework of [16], for proving termination of a CSRS on original terms, thus exploiting the given equivalence result. This approach has been implemented in the tool VMTL ([30])[1] and evaluated on a set of 24 examples. Several of these examples, where other existing approaches fail, could be shown to be operationally terminating thanks to the new method.[2]

For the sake of readability, only selected proofs will be presented inline. All other proofs can be found in the Appendix.

2 Preliminaries

We assume familiarity with the basic concepts and notations of term rewriting and context-sensitive rewriting (cf. e.g. [7], [8] and [20]). Throughout the paper we assume that all CTRSs, CSRSs and TRSs (i.e., their induced reduction relations) are *finitely branching*.

[1] `http://www.logic.at/vmtl/`

[2] First partial results of the current approach were presented at WST 2007, and some progress was reported at NWPT 2008.

By $Var(t)$ we denote the set of variables occurring in the term t. $Var^{\mu}(t)$ denotes the set of replacing variables and $\overline{Var^{\mu}}(t)$ the set of non-replacing variables w.r.t. a replacement map μ of t.

Conditional Rewriting We are concerned with *oriented* 3-CTRSs. Such systems consist of conditional rules $l \to r \Leftarrow c$, with c being of the form $s_1 \to^* t_1, \ldots, s_n \to^* t_n$ such that l is not a variable and $Var(r) \subseteq Var(l) \cup Var(c)$. The conditional rewrite relation induced by a CTRS $\mathcal{R}$ is inductively defined as follows: $R_0 = \emptyset$, $R_{j+1} = \{\sigma l \to \sigma r \mid l \to r \Leftarrow s_1 \to^* t_1, \ldots, s_n \to^* t_n \in \mathcal{R} \wedge \sigma s_i \to^*_{R_j} \sigma t_i$ for all $1 \leq i \leq n\}$, and $\to_{\mathcal{R}} = \bigcup_{j \geq 0} \to_{R_j}$. We say that a reduction step $s \to_{\mathcal{R}} t$ has depth i if $s \to_{\mathcal{R}_i} t$ and $s \not\to_{\mathcal{R}_j} t$ for all $j < i$. A deterministic CTRS (DCTRS) is an oriented 3-CTRS where for each rule $l \to r \Leftarrow s_1 \to t_1, \ldots, s_n \to t_n$ it holds that $Var(s_i) \subseteq Var(l) \cup \bigcup_{j=1}^{i-1} Var(t_j)$.

A DCTRS (Σ, R) is called *quasi-reductive*, cf. [29], [13], if there exists an extension Σ' of Σ and a well-founded partial order $\succ$ on $\mathcal{T}(\Sigma', V)$, which is monotonic, i.e., closed under contexts, such that for every rule $l \to r \Leftarrow s_1 \to^* t_1, \ldots, s_n \to^* t_n \in R$, every $\sigma \colon V \to \mathcal{T}(\Sigma', V)$ and every $i \in \{0, \ldots, n-1\}$:

- If $\sigma s_j \succeq \sigma t_j$ for every $1 \leq j \leq i$, then $\sigma l \succ_{st} \sigma s_{i+1}$.
- If $\sigma s_j \succeq \sigma t_j$ for every $1 \leq j \leq n$, then $\sigma l \succ \sigma r$.

Here $\succ_{st} = (\succ \cup \rhd)^+$ ($\rhd$ denotes the proper subterm relation).

A DCTRS $\mathcal{R} = (\Sigma, R)$ is *quasi-decreasing* [29] if there is a well-founded partial ordering $\succ$ on $\mathcal{T}(\Sigma, V)$, such that $\to_{\mathcal{R}} \subseteq \succ$, $\succ = \succ_{st}$, and for every rule $l \to r \Leftarrow s_1 \to^* t_1, \ldots, s_n \to^* t_n \in R$, every substitution σ and every $i \in \{0, \ldots, n-1\}$ it holds that $\sigma s_j \to^*_{\mathcal{R}} \sigma t_j$ for all $j \in \{1, \ldots, i\}$ implies $\sigma l \succ \sigma s_{i+1}$.

In [22] the notion of *operational termination* of (D)CTRSs is defined via the absence of infinite well-formed trees in a certain logical inference system. In the case of DCTRSs, this notion is shown to be equivalent to quasi-decreasingness [22].

The latter notions are related as follows ([29], [22]):

quasi-reductivity $\Rightarrow$ quasi-decreasingness $\Leftrightarrow$ operational termination

Context-Sensitive Narrowing and Orderings Given a CSRS $\mathcal{R} = (\Sigma, R)$ with replacement map μ, the relation of context-sensitive narrowing (written $\rightsquigarrow^{\mu}_{\mathcal{R}}$) is defined as $t \rightsquigarrow^{\mu}_{\mathcal{R}} s$ if there is a replacing non-variable position p in t such that $t|_p$ and l unify ($l \to r \in R$ and we assume that t and $l \to r$ do not share any variables) with mgu θ and $s = \theta(t[r]_p)$. We say that s is a *one-step, context-sensitive narrowing* of t. Note that in contrast to ordinary rewriting, here we allow for rules in R to have extra variables in the right-hand sides and variable left-hand sides. The reason for this general definition of narrowing is that we are going to use a *backward narrowing* relation that is induced by reversing all rules of a TRS (cf. Lemma 4 and Definition 11 below).

An ordering $\succ$ on terms $\mathcal{T}(\Sigma, V)$ is called μ-monotonic if f is monotonic in its i^{th} argument whenever $i \in \mu(f)$ for all $f \in \Sigma$, i.e.,

$$s_i \succ t_i \Rightarrow f(s_1, \ldots, s_{i-1}, s_i, s_{i+1} \ldots, s_n) \succ f(s_1, \ldots, s_{i-1}, t_i, s_{i+1} \ldots, s_n).$$

Context-Sensitive Dependency Pairs ([1, 2], cf. also [4]) Given a TRS $\mathcal{R} = (\Sigma, R)$, the signature Σ is partitioned into its defined and constructor symbols $\mathcal{D} \uplus \mathcal{C}$, where the defined symbols are exactly those that occur as root symbols of the left-hand sides of rules in R. A term t is *hidden* w.r.t. to a CSRS $(\mathcal{R} = ((\mathcal{D} \uplus \mathcal{C}, R), \mu))$ if $root(t) \in \mathcal{D}$ and t does not appear μ-replacing in a right-hand side of a rule of $\mathcal{R}$. Moreover, we say that a function f hides a position i if there is a rule $l \to r \in R$ such that some term $f(r_1, \ldots, r_i, \ldots, r_n)$ occurs at a non-replacing position of r, $i \in \mu(f)$ and r_i contains a defined symbol or a variable at a replacing position.

The set of context-sensitive dependency pairs ([1, 2]) of a CSRS $(\mathcal{R}, \mu)$, denoted $DP(\mathcal{R}, \mu)$, is $DP_o(\mathcal{R}, \mu) \cup DP_u(\mathcal{R}, \mu)$ where

$$DP_o(\mathcal{R}, \mu) = \{l^\sharp \to s^\sharp \mid l \to r \in R, r \trianglerighteq_\mu s, root(s) \in \mathcal{D}, l \not\trianglerighteq_\mu s\}$$

and $DP_u(\mathcal{R}, \mu)$ is the union of the following "unhiding" dependency pairs:

- $\{l^\sharp \to D^\sharp(x) \mid l \to r \in R, x \in Var^\mu(r) - Var^\mu(l)\}$
- $D^\sharp(f(x_1, \ldots, x_i, \ldots, x_n)) \to D^\sharp(x_i)$ for every function symbol f of any arity n and every $1 \leq i \leq n$ where f hides position i
- $D^\sharp(t) \to t^\sharp$ for every hidden term t

Here, $t^\sharp$ denotes the term $f^\sharp(t_1, \ldots, t_n)$, if $t = f(t_1, \ldots, t_n)$ and $f^\sharp$ is a new *dependency pair symbol*. Moreover, $D^\sharp$ is a fresh function symbol. The relation $\trianglerighteq_\mu$ is defined as $s \trianglerighteq_\mu t$ if $s = s[t]_p$ and $p \in Pos_\mu(t)$.

We denote by $\Sigma^\sharp$ the signature Σ plus all dependency pair symbols plus the new symbol $D^\sharp$. The replacement map μ is extended into $\mu^\sharp$ where $\mu^\sharp(f) = \mu(f)$, if $f \in \Sigma$, $\mu^\sharp(f^\sharp) = \mu(f)$, if $f^\sharp$ is a dependency pair symbol and $\mu(D^\sharp) = \emptyset$.

Let DP and $\mathcal{R}$ be TRSs and μ be a replacement map for their combined signature. A (possibly infinite) sequence of rules $s_1 \to t_1, s_2 \to t_2, \ldots$ from DP is a $(DP, \mathcal{R}, \mu)$-chain if there is a substitution σ, such that $\sigma t_i \to^*_{\mathcal{R},\mu} s_{i+1}$ for all $i > 0$. We say that σ *enables* the chain $s_1 \to t_1, s_2 \to t_2, \ldots$

We call a triple $(DP, \mathcal{R}, \mu)$, where DP and $\mathcal{R}$ are TRSs and μ is a replacement map for the combined signatures of DP and $\mathcal{R}$, a *(context-sensitive) dependency pair problem* (CS-DP-problem). A context-sensitive dependency pair problem is finite if there is no infinite $(DP, \mathcal{R}, \mu)$-chain.

A CSRS $(\mathcal{R}, \mu)$ is μ-terminating if and only if the dependency pair problem $(DP(\mathcal{R}, \mu), \mathcal{R}, \mu)$ is finite ([1, 2]).

3 Context-Sensitive Quasi-Reductivity

The goal of this work is to provide methods for proving *operational termination* of DCTRSs. We define the notion of context-sensitive quasi-reductivity, which is equivalent to operational termination (cf., Corollary 3 below), and the key to several main results of this paper.

Definition 1 (context-sensitive quasi-reductivity). *A DCTRS $\mathcal{R}$ ($\mathcal{R} = (\Sigma, R)$) is called* context-sensitively quasi-reductive *(*cs-quasi-reductive*) if there is an extension of the signature Σ' ($\Sigma' \supseteq \Sigma$), a replacement map μ (s.t. $\mu(f) = \{1, \ldots, ar(f)\}$ for all $f \in \Sigma$) and a μ-monotonic, well-founded partial order $\succ_\mu$ on $\mathcal{T}(\Sigma', V)$ satisfying for every rule $l \to r \Leftarrow s_1 \to^* t_1, \ldots, s_n \to^* t_n$, every substitution $\sigma : V \to \mathcal{T}(\Sigma, V)$ and every $i \in \{0, \ldots, n-1\}$:*

- *If $\sigma s_j \succeq_\mu \sigma t_j$ for every $1 \leq j \leq i$ then $\sigma l \succ_\mu^{st} \sigma s_{i+1}$.*
- *If $\sigma s_j \succeq_\mu \sigma t_j$ for every $1 \leq j \leq n$ then $\sigma l \succ_\mu \sigma r$.*

The ordering $\succ_\mu^{st}$ is defined as $(\succ_\mu \cup \rhd_\mu)^+$ where $t \rhd_\mu s$ if and only if s is a proper subterm of t at some position $p \in Pos^\mu(t)$. Moreover $\succeq = (\succ \cup =)$.

To be entirely precise, the notion of cs-quasi-reductivity should be parameterized by the set of function symbols that may not be restricted by the replacement map μ. However, as throughout the paper this set of function symbols is the set of functions of the signature of the DCTRS in question, we refrain from giving a reference to this parameter in the notion *cs*-quasi-reductivity for the sake of simplicity.

Cs-quasi-reductivity generalizes quasi-reductivity in the sense that the extended signature may be equipped with a replacement map (which must leave the original signature untouched, though) and the monotonicity requirement of the ordering is relaxed accordingly. Furthermore, and this is crucial, in the ordering constraints for the conditional rules the substitutions replace variables only by terms over the original signature, whereas in the original definition (of quasi-reductivity) terms over the extended signature are substituted.

The latter generalization appears to be quite natural, since the main implications of quasi-reductivity remain valid (cf. Proposition 2). Moreover, it is the key to the completeness results that we will prove (cf. Corollary 3).

Proposition 1. *If a DCTRS $\mathcal{R}$ is quasi-reductive, then it is cs-quasi-reductive.*

Proposition 2. *If a DCTRS $\mathcal{R}$ is cs-quasi-reductive, then it is quasi-decreasing.*

Corollary 1. *Let $\mathcal{R}$ be DCTRS. If $\mathcal{R}$ is cs-quasi-reductive, then it is* operationally terminating.

4 Proving Context-Sensitive Quasi-Reductivity

In the following, we use a transformation from DCTRSs into CSRSs such that μ-termination of the transformed CSRS implies cs-quasi-reductivity of the original

DCTRS. The transformation is actually a variant of the one in [29], which in turn was inspired by [24, 25].[3]

Definition 2 (unraveling of DCTRSs, [29])**.** *Let $\mathcal{R}$ be a DCTRS ($\mathcal{R} = (\Sigma, R)$). For every rule $\alpha : l \to r \Leftarrow s_1 \to^* t_1, \ldots, s_n \to^* t_n$ we use n new function symbols U_i^α ($i \in \{1, \ldots, n\}$). Then α is transformed into a set of unconditional rules in the following way:*

$$\begin{array}{rcl} l & \to & U_1^\alpha(s_1, Var(l)) \\ U_1^\alpha(t_1, Var(l)) & \to & U_2^\alpha(s_2, Var(l), \mathcal{E}Var(t_1)) \\ & \vdots & \\ U_n^\alpha(t_n, Var(l), \mathcal{E}Var(t_1), & & \\ \ldots, \mathcal{E}Var(t_{n-1})) & \to & r \end{array}$$

Here $Var(s)$ denotes an arbitrary but fixed sequence of the set of variables of the term s. Let $\mathcal{E}Var(t_i)$ be $Var(t_i) \setminus (Var(l) \cup \bigcup_{j=1}^{i-1} Var(t_j))$. As before, abusing notation, by $\mathcal{E}Var(t_i)$ we mean an arbitrary but fixed sequence of the variables in the set $\mathcal{E}Var(t_i)$. Any unconditional rule of $\mathcal{R}$ is transformed into itself. The transformed system $U(\mathcal{R}) = (U(\Sigma), U(R))$ is obtained by transforming each rule of $\mathcal{R}$ where $U(\Sigma)$ is Σ extended by all new function symbols. In case $\mathcal{R}$ has only one conditional rule α, we also write U_i instead of U_i^α.

Henceforth, we use the notion of U-symbols of a transformed signature, which are function symbols from $U(\Sigma) \setminus \Sigma$. Moreover, by U-terms or U-rooted terms we mean terms with a U-symbol as their root.

Next, we define the function *tb*, whose intended meaning is to undo non-finished meta-evaluations, i.e., evaluations of the form $s \to^*_{U(\mathcal{R})} U(v_1, ..., v_l)$. We call reductions of this shape meta-evaluations, because they are used for the evaluation of encoded conditions. This evaluation does not have an explicit counterpart in conditional rewrite sequences. The function *tb* and its properties will play a crucial role in understanding and proving the main results of this paper.

Definition 3. *The mapping $tb\colon \mathcal{T}(U(\Sigma), V) \to \mathcal{T}(\Sigma, V)$ (read "translate back") which is equivalent to Ohlebusch's mapping ∇ ([29, Definition 7.2.53]) is defined by*

$$tb(t) = \begin{cases} x & \text{if } t = x \in V \\ f(tb(v_1), \ldots, tb(v_l)) & \text{if } t = f(v_1, \ldots, v_l) \\ & \quad \text{and } f \in \Sigma \\ \sigma l & \text{if } t = U_j^\alpha(v_1, \ldots, v_{m_j}) \\ & \quad \text{and } \alpha = l \to r \Leftarrow c \end{cases}$$

where $Var(l) = x_1, \ldots, x_k$ and σ is defined as $\sigma x_i = tb(v_{i+1})$ for $1 \leq i \leq k$. Note that from Definition 2 it follows that $m_j \geq k + 1$.

[3]Note that there exist also various other transformations from conditional to unconditional TRSs in the literature, cf. e.g. [5], [31] and [17] for more recent ones. However, for our purposes, the chosen transformation appears to be the most appropriate one.

Informally, the mapping tb translates back an evaluation of conditions to its start. Thus, $tb(u) = u$ for every term $u \in \mathcal{T}(\Sigma, V)$. Note that in general $s = tb(t)$ does not imply $s \to^*_{U(\mathcal{R})} t$. The reason is that, for a term $t = U^\alpha_j(v_1, \ldots, v_l)$, the definition of $tb(t)$ completely ignores the first argument t_1 of U^α_j.

Example 1. *Let $\mathcal{R}$ be a DCTRS consisting of one rule*

$$f(x) \quad \to \quad a \Leftarrow x \to b$$

$U(\mathcal{R})$ is given by the two rules

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

Consider the term $t = U(a, b)$. We have $tb(t) = f(b)$ and clearly $f(b) \not\to^*_{U(\mathcal{R})} U(a, b)$.

Informally, the term $t = U^\alpha_j(v_1, \ldots, v_{m_j})$ represents an intermediate state of a reduction in $U(\mathcal{R})$ issuing from an original term, i.e., a term from $\mathcal{T}(\Sigma, V)$, only if v_1 can be obtained (by reduction in U($\mathcal{R}$)) from the corresponding instance of the left-hand side of the corresponding condition of the applied conditional rule α.

The transformation of Definition 2 is sound w.r.t. quasi-reductivity, i.e., whenever the transformed system $U(\mathcal{R})$ is terminating, the original DCTRS $\mathcal{R}$ is *quasi-reductive* [29]. The transformation is not complete in this respect, though.

Example 2. *([24]) Consider the DCTRS $\mathcal{R} = (\Sigma, R)$ given by*

$$\begin{array}{llr} a \to c & c \to l & h(x, x) \to g(x, x, f(k)) \\ a \to d & d \to m & g(d, x, x) \to A \\ b \to c & k \to l & A \to h(f(a), f(b)) \\ b \to d & k \to m & \alpha : f(x) \to x \Leftarrow x \to^* e \\ c \to e & & \end{array}$$

The system $U(\mathcal{R}) = (U(\Sigma), U(R))$ is given by $U(\Sigma) = \Sigma \cup \{U^\alpha_1\}$ and $U(R) = R$ except that rule α is replaced by the rules $f(x) \to U^\alpha_1(x, x)$ and $U^\alpha_1(e, x) \to x$. $\mathcal{R}$ is operationally terminating (cf. Example 4, below), nevertheless $U(\mathcal{R})$ is non-terminating ([29]).

Roughly speaking, the problem in Example 2 is that subterms at the second position of U^α_1 are reduced, which is actually only supposed to "store" the variable bindings for future rewrite steps. These reductions can be prevented by using context-sensitivity. More precisely, we intend to forbid reductions of subterms which occur at or below a second, third, etc. argument position of an auxiliary U-symbol, according to the intuition that during the evaluation of conditions, the variable bindings should remain untouched. This leads to the following modification of the transformation, which has already been proposed independently by several authors (e.g., [11], [27], [12]).

Definition 4. *(context-sensitive unraveling of a DCTRS) Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS. The context-sensitive rewrite system $U_{cs}(\mathcal{R}) = (U(\Sigma), U(R))$ uses the same signature and the same rules as $U(\mathcal{R})$. Additionally, a replacement map $\mu_{U_{cs}(\mathcal{R})}$ is used with $\mu_{U_{cs}(\mathcal{R})}(U) = \{1\}$ if $U \in U(\Sigma) \setminus \Sigma$ and $\mu_{U_{cs}(\mathcal{R})}(f) = \{1, \ldots, ar(f)\}$ if $f \in \Sigma$.*

For notational simplicity we refer to $\mu_{U_{cs}(\mathcal{R})}$ just as μ if no confusion arises, e.g. in "μ-termination of $U_{cs}(\mathcal{R})$".

Note that context-sensitivity assures that in a reduction of the form

$$\begin{array}{rll} U_i(\sigma_i' s_i, \sigma_i \vec{x}_i) & \xrightarrow{>\epsilon *}_{U_{cs}(\mathcal{R})} & U_i(\sigma_i'' t_i, \sigma_{i+1}\vec{x}_i) \\ & \xrightarrow{\epsilon}_{U_{cs}(\mathcal{R})} & U_{i+1}(\sigma_{i+1}' s_{i+1}, \sigma_{i+1}\vec{x}_{i+1}) \end{array}$$

where $\vec{x}_i$ (resp. $\vec{x}_{i+1}$) denotes the sequence $x_1, \ldots x_{k_i}$ (resp. $x_1, \ldots x_{k_{i+1}}$) of variables, σ_i and σ_{i+1} are not contradictory, i.e., $\sigma_i x = \sigma_{i+1} x$ for all $x \in Dom(\sigma_i) \cap Dom(\sigma_{i+1})$. In fact this is a crucial property of $U_{cs}(\mathcal{R})$, because given a DCTRS $\mathcal{R} = (\Sigma, R)$ it guarantees that for each term $t \in \mathcal{T}(U(\Sigma), V)$ we have that $tb(t) \rightarrow^*_{U_{cs}(\mathcal{R})} t$ provided that t is reachable by *any* term $s \in \mathcal{T}(\Sigma, V)$ (see Lemma 1, below). This is in general not true, if context-sensitivity is dropped.

Example 3. *Let $\mathcal{R} = (\Sigma, R)$ be the DCTRS of Example 1 extended by two unconditional rules*

$$\begin{array}{rcl} f(x) & \rightarrow & a \Leftarrow x \rightarrow b \\ a & \rightarrow & b \\ a & \rightarrow & c \end{array}$$

The transformed system $U(\mathcal{R})$ is

$$\begin{array}{rcl} f(x) & \rightarrow & U(x, x) \\ U(b, x) & \rightarrow & a \\ a & \rightarrow & b \\ a & \rightarrow & c \end{array}$$

Consider the term $t = U(b, c)$. It is reachable in $U(\mathcal{R})$ from $f(a) \in \mathcal{T}(\Sigma, V)$:

$$f(a) \rightarrow_{U(\mathcal{R})} U(a, a) \rightarrow_{U(\mathcal{R})} U(b, a) \rightarrow_{U(\mathcal{R})} U(b, c)$$

However, it is obviously not reachable from $tb(t) = f(c)$ as b is not reachable from c. On the other hand, within $U_{cs}(\mathcal{R})$, $U(b, c)$ is not reachable by any *term from $\mathcal{T}(\Sigma, V)$ because in $U_{cs}(\mathcal{R})$ reachability of a term t by any term $s \in \mathcal{T}(\Sigma, R)$ coincides with reachability of t from $tb(t)$.*

The fact that in a CSRSs $U_{cs}(\mathcal{R})$, obtained by the context-sensitive transformation after transforming a DCTRS $\mathcal{R} = (\Sigma, R)$, each term t is reachable from $tb(t)$ if t is part of reduction sequence issuing from a term of $\mathcal{T}(\Sigma, V)$, will be used extensively in the proofs of the main results of this paper.

Before investigating the effects of this modification on the power of proving operational termination, let us consider the simulatability of a DCTRS $\mathcal{R}$ by $U_{cs}(\mathcal{R})$. While *simulation completeness*, i.e., the property of $U_{cs}(\mathcal{R})$ being able to mimic reductions of $\mathcal{R}$, is easy to obtain, *simulation soundness*, i.e., the property of $U_{cs}(\mathcal{R})$ to allow *only* those reductions (from original terms to original terms) that are also possible in $\mathcal{R}$, is non-trivial.

In [27] it was shown that simulation soundness is obtained if an additional restriction is imposed on reductions in $U_{cs}(\mathcal{R})$, which roughly states that only redexes without U-symbols (except at the root position) may be contracted. However, this additional "membership condition" is not really needed.

Theorem 1 (simulation completeness)**.** *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS. For every $s, t \in \mathcal{T}(\Sigma, V)$ we have: If $s \to_{\mathcal{R}} t$, then $s \to^+_{U_{cs}(\mathcal{R})} t$.*

Theorem 2 (simulation soundness)**.** *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS. For every $s, t \in \mathcal{T}(\Sigma, V)$ we have: If $s \to^+_{U_{cs}(\mathcal{R})} t$, then $s \to^+_{\mathcal{R}} t$.*

Proof. Before proving the result we define the U-depth of a term s as the maximal number m of positions $p_1 < ... < p_m$ such that $s|_{p_i}$ is a U-term for all i. The U-depth of a reduction $s_1 \to ... \to s_n$ is the maximal U-depth of the s_i, $1 \leq i \leq n$. Note that there exists a reduction $tb(s) \to^* s$ for a term $s \in \mathcal{T}(U(\Sigma), V)$ with the same U-depth as s, if such a reduction exists at all.

Now, we prove the theorem by proving (the more general fact) that $s \to^*_{U_{cs}(\mathcal{R})} t$ ($s, t \in \mathcal{T}(U(\Sigma), V)$) implies $tb(s) \to^*_{\mathcal{R}} tb(t)$ provided that s is reachable from an original term, i.e., $tb(s) \to^*_{U_{cs}(\mathcal{R})} s$ (we prove non-emptiness of the simulating reduction sequence afterwards). We use induction on the U-depth of the derivation $s \to^*_{U_{cs}(\mathcal{R})} t$. We assume that this reduction is non-empty. Otherwise the claim holds trivially.

If the U-depth is 0, all terms of this reduction are terms over the original signature. Thus, only unconditional rules of $U(R) \cap R$ have been used. Furthermore, $s = tb(s)$ and $t = tb(t)$ and $tb(s) \to^*_{\mathcal{R}} tb(t)$ and the latter reduction is non-empty if $s \to^*_{U_{cs}(\mathcal{R})} t$ is.

Next, if the U-depth of $s \to^*_{U_{cs}(\mathcal{R})} t$ is greater than 0, we will show that the reduction sequence can be simulated in $\mathcal{R}$ step by step maintaining for each term s' in $s \to^*_{U_{cs}(\mathcal{R})} t$ the invariant

$$s \quad \to^*_{U_{cs}(\mathcal{R})} \quad s' \qquad\qquad tb(s) \quad \to^*_{\mathcal{R}} \quad tb(s').$$

Thus, s' is reachable by an original term (because s is), therefore $tb(s') \to^*_{U_{cs}(\mathcal{R})} s'$.

Now, consider a first step $s \to_{U_{cs}(\mathcal{R})} s'$ of $s \to^*_{U_{cs}(\mathcal{R})} t$. It has a U-depth of at least the U-depth of s. If this step is tb-preserving, then $tb(s) = tb(s')$ and s' trivially satisfies the invariant. Otherwise we distinguish two cases:

1. First, assume an unconditional rule (from $U(R) \cap R$) is applied at $s|_p = \sigma l$. As the step is not *tb*-preserving, there is no more outer U-symbol in s. Thus, $tb(s)|_p = \sigma' l$ and $\sigma' x = tb(\sigma x)$ for all $x \in Dom(\sigma)$. Hence, $tb(s) \to_{\mathcal{R}} tb(s)[\sigma' r]_p = tb(s')$.

2. Second, assume a rule $l' = U_n^\alpha(t_n, x_1, ..., x_m) \to r$ is applied where $\alpha = l \to r \Leftarrow s_1 \to t_1, ..., s_n \to t_n \in R$. Say $s|_p = \sigma l'$. The reduction $tb(s) \to^*_{U_{cs}(\mathcal{R})} s$ has a U-depth which is less than or equal to the U-depth of s. Furthermore, each reduction $\sigma s_i \to^*_{U_{cs}(\mathcal{R})} \sigma t_i$, $1 \leq i \leq n$, occurs as subreduction of the former one (i.e. $tb(s) \to^*_{U_{cs}(\mathcal{R})} s$). Moreover, each of the latter reductions takes place inside a U-term, thus the U-depth of these reductions (extracted from their contexts) is smaller and the induction hypothesis applies yielding $tb(\sigma s_i) = \sigma' s_i \to^*_{\mathcal{R}} tb(\sigma t_i) = \sigma' t_i$ for all $1 \leq i \leq n$. Consequently, $tb(s) \to_{\mathcal{R}} tb(s)[\sigma' r] = tb(s')$.

Hence, $tb(s) \to^*_{\mathcal{R}} tb(t)$ follows by repetition. Note that each non-*tb*-preserving step in $s \to^*_{U_{cs}(\mathcal{R})} t$ corresponds to a non-empty step in $tb(s) \to^*_{\mathcal{R}} tb(t)$. Whenever $s \to^+_{U_{cs}(\mathcal{R})} t$ and $s, t \in \mathcal{T}(\Sigma, V)$, there is at least one non-*tb*-preserving step in this reduction. Thus we obtain $tb(s) = s \to^+_{\mathcal{R}} tb(t) = t$. □

Regarding termination, the transformation of Definition 4 is sound for *cs*-quasi-reductivity in the sense that μ-termination of $U_{cs}(\mathcal{R})$ implies context-sensitive quasi-reductivity and thus operational termination of $\mathcal{R}$.

Theorem 3 (sufficiency for cs-quasi-reductivity)**.** *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS. If $U_{cs}(\mathcal{R})$ is μ-terminating, then $\mathcal{R}$ is cs-quasi-reductive.*

Proof. As $U_{cs}(\mathcal{R})$ is $\mu_{U_{cs}(\mathcal{R})}$-terminating, $\succ_\mu = \to^+_{U_{cs}(\mathcal{R})}$ is a μ-reduction ordering on $\mathcal{T}(U(\Sigma), V)$ (where $U(\Sigma) \supseteq \Sigma$). Assume $\sigma s_j \succeq_\mu \sigma t_j$ for every $1 \leq j \leq i < n$ for a rule $\alpha : l \to r \Leftarrow s_1 \to^* t_1, ..., s_n \to^* t_n$ ($\sigma : V \to \mathcal{T}(\Sigma, V)$). Then we have the following sequence in $U_{cs}(\mathcal{R})$:

$$\begin{array}{lll}
\sigma l & \to_{U_{cs}(\mathcal{R})} & \sigma U_1^\alpha(s_1, Var(l)) \\
 & \to^*_{U_{cs}(\mathcal{R})} & \sigma U_1^\alpha(t_1, Var(l)) \\
 & \to_{U_{cs}(\mathcal{R})} & \sigma U_2^\alpha(s_2, Var(l), \mathcal{E}Var(t_1)) \\
 & \to^*_{U_{cs}(\mathcal{R})} & \sigma U_2^\alpha(t_2, Var(l), \mathcal{E}Var(t_1)) \\
\dots & & \\
 & \to_{U_{cs}(\mathcal{R})} & \sigma U_i^\alpha(s_i, Var(l), \mathcal{E}Var(t_1), ..., \mathcal{E}Var(t_{i-1})) \\
 & \to^*_{U_{cs}(\mathcal{R})} & \sigma U_i^\alpha(t_i, Var(l), \mathcal{E}Var(t_1), ..., \mathcal{E}Var(t_{i-1})) \\
 & \to_{U_{cs}(\mathcal{R})} & \sigma U_{i+1}^\alpha(s_{i+1}, Var(l), \mathcal{E}Var(t_1), ..., \mathcal{E}Var(t_i))
\end{array}$$

Thus $\sigma l \succ^{st}_\mu \sigma s_{i+1}$. If $\sigma s_j \succeq_\mu \sigma t_j$ for all $1 \leq j \leq n$, then it is easy to see that there is a reduction sequence $\sigma l \to^+_{U_{cs}(\mathcal{R})} \sigma r$, thus $\sigma l \succ_\mu \sigma r$. □

The following corollary has already been proved in [12].

Corollary 2. *([12]) Let $\mathcal{R}$ be a DCTRS. If $U_{cs}(\mathcal{R})$ is μ-terminating, then $\mathcal{R}$ is operationally terminating.*

Obviously, as $U(\mathcal{R})$ and $U_{cs}(\mathcal{R})$ differ only in that $U_{cs}(\mathcal{R})$ uses an additional replacement map, the context-sensitive transformation is more powerful when it comes to verifying operational termination.

Proposition 3. *([12]) Let $\mathcal{R}$ be a DCTRS. If $U(\mathcal{R})$ is terminating, then $U_{cs}(\mathcal{R})$ is μ-terminating.*

Example 4. *Consider the DCTRS $\mathcal{R}$ of Example 2. The transformed system $U_{cs}(\mathcal{R})$ (which is identical to $U(\mathcal{R})$, except for the fact that an additional replacement map is used) is μ-terminating. This can for instance be proved by minimal counterexample and case analysis. However, we will see that in order to verify operational termination of $\mathcal{R}$, it is sufficient to prove a weaker form of termination, which can be handled automatically (see Theorem 5 and Example 9 below).*

Unfortunately, and interestingly, cs-quasi-reductivity of a DCTRS $\mathcal{R}$ does not imply μ-termination of $U_{cs}(\mathcal{R})$.

Example 5. *([29, Ex. 7.2.51]) Consider the DCTRS $\mathcal{R}$ given by*

$$\begin{array}{rcl} g(x) & \to & k(y) \Leftarrow h(x) \to^* d, h(x) \to^* c(y) \\ h(d) & \to & c(a) \\ h(d) & \to & c(b) \\ f(k(a), k(b), x) & \to & f(x, x, x) \end{array}$$

This system is quasi-reductive (and thus cs-quasi-reductive) (cf., [29]). However, the system $U_{cs}(\mathcal{R})$, where the conditional rule is replaced by

$$\begin{array}{rcl} g(x) & \to & U_1(h(x), x) \\ U_1(d, x) & \to & U_2(h(x), x) \\ U_2(c(y), x) & \to & k(y) \end{array}$$

with $\mu(U_i) = \{1\}$ for $i \in \{1, 2\}$, is not μ-terminating.

$$\begin{array}{ll} & f(k(a), k(b), U_2(h(d), d)) \\ \to_{U_{cs}(\mathcal{R})} & f(U_2(h(d), d), U_2(h(d), d), U_2(h(d), d)) \\ \to^+_{U_{cs}(\mathcal{R})} & f(U_2(c(a), d), U_2(c(b), d), U_2(h(d), d)) \\ \to^+_{U_{cs}(\mathcal{R})} & f(k(a), k(b), U_2(h(d), d)) \end{array}$$

Note that in this counterexample the crucial subterm $t' = U_2(h(d), d)$ which reduces to both $k(a)$ and $k(b)$ does not have a counterpart in the original system, i.e., a term $t \in \mathcal{T}(\Sigma, V)$ with $t \to^*_{U_{cs}(\mathcal{R})} t'$. Hence, it seems natural to conjecture that such counterexamples are impossible if we only consider derivations issuing from original terms. This is indeed the case, even for quasi-decreasing systems (cf. Theorems 4 and 5 below). Before proving these results we need some additional machinery, though.

Definition 5 (μ-termination on original terms)**.** *A CSRS $\mathcal{R} = (U(\Sigma), U(R))$ with replacement map μ, obtained by the transformation of Definition 4 is called* μ-terminating on original terms*, if there is no infinite reduction chain issuing from a term $t \in \mathcal{T}(\Sigma, V)$ in $\mathcal{R}$.*

Certain reduction steps inside a U-term t will have no effect on the result of the function tb, i.e., $t \to s$ and $tb(t) = tb(s)$. The following definition identifies those reductions. First, obviously reductions that occur strictly inside a U-term t do not alter the result of tb. The reason is that because of context-sensitivity these reductions can only take place in the first argument of the root U-symbol and furthermore according to the definition of tb this first argument is irrelevant for the computation of tb.

Second, if a rule of the form $U^i_\alpha(s_1, ..., s_n) \to U^{i+1}_\alpha(s_1, ..., s_n)$ (whose right-hand side is a U-term) is applied to t then tb applied to the resulting term yields also the same result as $tb(t)$. The reason is that the variable bindings inside the U-term are preserved in such a step and all the variables that are present in l (where $\alpha = l \to r \Leftarrow c$) are already bound. For the same reason $tb(t) = tb(s)$ if t is not a U-term, s is a U-term and $t \to s$.

Definition 6 (tb-preserving reduction steps)**.** *Let $\mathcal{R}$ be a DCTRS ($\mathcal{R} = (\Sigma, R)$) and $U_{cs}(\mathcal{R}) = (U(\Sigma), U(R))$ its transformed CSRS. A step $s \xrightarrow{p}_{U_{cs}(\mathcal{R})} t$*[4] *is called* tb-preserving *if either p is strictly below some position q of s, where $root(s|_q)$ is a U-symbol, or $(t|_p)$ is a U-term.*

The intuition behind tb-preserving steps is that whenever $s \to_{U_{cs}(\mathcal{R})} t$ with a tb-preserving step, we have $tb(s) = tb(t)$.

Example 6. *Consider a CSRS $\mathcal{R}$*

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

with $\mu(U) = \mu(f) = \{1\}$. The following reductions are tb-preserving:

$$\begin{array}{rcl} \underline{f(a)} & \to_\mu & U(b, a),\ as\ tb(f(a)) = tb(U(b, a)) = f(a) \\ U(\underline{b}, a) & \to_\mu & U(c, a),\ as\ tb(U(b, a)) = tb(U(c, a)) = f(a) \end{array}$$

while this one is not:

$$\underline{U(c, a)} \quad \to_\mu \quad a,\ due\ to\ tb(U(c, a)) = f(a) \neq tb(a) = a$$

The following "commutation" lemma allows us to commute reduction steps that occur as part of a "meta evaluation" to check truth or falsity of conditions with other "normal" reductions.

[4] $\xrightarrow{p}$ denotes a reduction step at position p.

Lemma 1. *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS and let $U_{cs}(\mathcal{R})$ be its transformed system. For every term $s \in \mathcal{T}(\Sigma, V)$ and every reduction $s \to^*_{U_{cs}(\mathcal{R})} t$ there is a reduction sequence $s \to^*_{U_{cs}(\mathcal{R})} tb(t) \to^*_{U_{cs}(\mathcal{R})} t$, such that each reduction step in the subsequence from $tb(t)$ to t is tb-preserving.*

Proof. From the proof of Proposition 2 we know that $s \to^*_{\mathcal{R}} tb(t)$. Moreover, Proposition 1 yields $s \to^*_{U_{cs}(\mathcal{R})} tb(t)$. Each U-term u in t can be traced back to the point where its U-root symbol (or a predecessor U-symbol) was first introduced by a rule having a left-hand side containing only original symbols. The reason is that U-symbols do not occur anywhere in the left-hand sides of the rewrite rules of $U_{cs}(\mathcal{R})$ but at the root position and therefore U-terms can be traced back in $U_{cs}(\mathcal{R})$-reductions. Thus, all reduction steps necessary to reduce $tb(u)$ to u already occurred in the reduction sequence $s \to^*_{U_{cs}(\mathcal{R})} t$ (not necessarily one after the other) and thus we have $tb(u) \to^*_{U_{cs}(\mathcal{R}),\mu} u$ for all such terms u and finally $tb(t) \to^*_{U_{cs}(\mathcal{R}),\mu} t$. Note that all these steps introduce or preserve U-symbols, thus they are tb-preserving. $\square$

Now we can state the main results of this section.

Theorem 4. *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS. If $\mathcal{R}$ is quasi-decreasing, then $U_{cs}(\mathcal{R})$ is μ-terminating on $\mathcal{T}(\Sigma, V)$.*

Proof. For notational simplicity in the sequel we write $\to$ instead of $\to_{U_{cs}(\mathcal{R})}$. For a proof by minimal counterexample suppose that $s \in \mathcal{T}(\Sigma, V)$ initiates an infinite $\to$-reduction $D\colon s \to \dots$ such that there is no $s' \in \mathcal{T}(\Sigma, V)$, $s \succ s'$ with this property (where $\succ$ is the quasi-decreasing ordering). Since $\succ$ contains the subterm ordering, this implies that every proper subterm of s is $\to$-terminating. Hence, D must have at least one root reduction step, i.e., be of the shape $s \to^* t \xrightarrow{\epsilon} u \to \dots$ where $t \xrightarrow{\epsilon} u$ is the first root reduction step. Since the root symbol of s is from the original signature, the left-hand side of the rule applied to t must be a term of the original signature. There are two possibilities now.

First, assume an unconditional rule $l \to r$ ($l, r \in \mathcal{T}(\Sigma, V)$ was applied to t. Then, $t = \sigma l$, $u = \sigma r$. According to Lemma 1 we have $s \to^* tb(t) \to^* t$. Since $t = \sigma l$, we get $tb(t) = \sigma' l$, because the steps from $tb(t)$ to t are tb-preserving and $\sigma' x \to^* \sigma x$ for all $x \in Dom(\sigma)$. Thus, we have $s \to^* tb(t) = \sigma' l \to \sigma' r \to^* \sigma r = u$. Furthermore, by quasi-decreasingness we get $s \succ \sigma' r$ because of $\to_{\mathcal{R}} \subseteq \succ$ and $s \to^+ \sigma' r \Rightarrow s \to^+_{\mathcal{R}} \sigma' r \in \mathcal{T}(\Sigma, V)$ (according to Theorem 2). This means that $\sigma' r \prec s$ also initiates an infinite $\to$-reduction, hence yields a smaller counterexample. But this contradicts our minimality assumption.

Secondly, assume the transformed version of a conditional rule $l \to r \Leftarrow s_1 \to^* t_1, \dots, s_n \to^* t_n$ is applied to t. Hence, $t = \sigma l$ and as before we get $tb(t) = \sigma' l$ where $\sigma' x \to^* \sigma x$ for all $x \in Dom(\sigma)$. Thus $u = \sigma U_1(s_1, x_1, \dots, x_{k_1})$ and we have $tb(t) \to \sigma' U_1(s_1, x_1, \dots, x_{k_1})$. By quasi-decreasingness we get $\sigma' l \succ \sigma' s_1, \sigma' x_1, \dots, \sigma' x_{k_1}$, hence all the latter terms are terminating by minimality of the counterexample. Therefore, σs_1 and $\sigma x_1, \dots, \sigma x_{k_1}$ are terminating, too, because of $\sigma' y \to^* \sigma y$ for all $y \in Dom(\sigma)$. Thus, the only possibility of an infinite reduction from u is via a next root reduction step:

$u = \sigma U_1(s_1, x_1, \ldots, x_{k_1}) \to^* \sigma_1 U_1(t_1, x_1, \ldots, x_{k_1}) \xrightarrow{\epsilon} \sigma_1 U_2(s_2, x_1, \ldots, x_{k_2})$. So $\sigma' s_1 \to^* \sigma s_1 \to^* \sigma_1 t_1$, and Lemma 1 yields $\sigma' s_1 \to^* tb(\sigma_1 t_1) = \sigma'_1 t_1 \to^* \sigma_1 t_1$. Then it also holds that $\sigma'_1 U_1(t_1, x_1, \ldots, x_{k_1}) \to \sigma'_1 U_2(s_2, x_1, \ldots, x_{k_2})$ and as $\sigma'_1 s_1 \to^* \sigma'_1 t_1$, we have $\sigma'_1 s_1 \to^*_{\mathcal{R}} \sigma'_1 t_1 \in \mathcal{T}(\Sigma, V)$ according to Theorem 2 and thus $\sigma'_1 l \succ \sigma'_1 s_2$. By minimality, $\sigma'_1 s_2$ and $\sigma'_1 x_1, \ldots, \sigma'_1 x_{k_2}$ are terminating, hence also $\sigma_1 s_2$ and $\sigma_1 x_1, \ldots, \sigma_1 x_{k_2}$ because of $\sigma' \to^* \sigma$. Similarly, an infinite reduction from $\sigma_1 U_2(s_2, x_1, \ldots, x_{k_2})$ is only possible via a next reduction step for which we need $\sigma_1 s_2 \to^* \sigma_2 t_2$ for some σ_2. By continuing this argumentation, we finally get that σl must eventually be reduced to $\sigma_n U_n(t_n, x_1, \ldots, x_{k_n})$ and $\sigma' l$ can be reduced to $\sigma'_n U(t_n, x_1, \ldots, x_{k_n})$. We have that $\sigma'_n t_n \in \mathcal{T}(\Sigma, V)$ is terminating by minimality (and quasi-decreasingness) and $\sigma_n t_n$ is terminating because of $\sigma'_n t_n \to^* \sigma_n t_n$. Therefore, the term $\sigma_n U(t_n, x_1, \ldots, x_{k_n})$ is reduced to $\sigma_n r$ and $\sigma'_n U(t_n, x_1, \ldots, x_{k_n})$ can be reduced to $\sigma'_n r$. We have $\sigma' l (= \sigma'_n l) \succ \sigma'_n r$ because of $\sigma' l \to^+ \sigma'_n r \in \mathcal{T}(\Sigma, V)$ and thus $\sigma' l \to^+_{\mathcal{R}} \sigma'_n r$ by Theorem 2. Hence, $\sigma'_n r$ (with $s \to^* \sigma'_n r \to^* \sigma_n r$) is terminating because of minimality and $\sigma_n r$ is terminating due to $\sigma'_n r \to^* \sigma_n r$. But this contradicts the counterexample property (of s). Hence, we are done. $\square$

Conversely, cs-quasi-reductivity follows from termination of the transformed system on original terms.

Theorem 5. *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS. If $U_{cs}(\mathcal{R})$ is μ-terminating on $\mathcal{T}(\Sigma, V)$, then $\mathcal{R}$ is cs-quasi-reductive.*

Proof. We define the ordering $\succ$ by $s \succ t$ if $s \to^+_{U_{cs}(\mathcal{R})} t$ and s is reachable (in $\to_{U_{cs}(\mathcal{R})}$) by a term of the original signature (i.e. $tb(s) \to^*_{U_{cs}(\mathcal{R})} s$). This relation is well-founded because $\to_{U_{cs}(\mathcal{R})}$ is terminating on $\mathcal{T}(\Sigma, V)$. Let $\succ_\mu$ be the μ-monotonic closure of $\succ$ w.r.t. $\mathcal{T}(U(\Sigma), V)$, i.e., $C[s]_p \succ_\mu C[t]_p$ if $s \succ t \wedge p \in Pos^\mu(C[s]_p)$. We show that $\mathcal{R}$ is cs-quasi-reductive w.r.t. $\succ_\mu$. Note that $\succ_\mu \subseteq \to^+_{U_{cs}(\mathcal{R})}$.

First, we will deal with well-foundedness of $\succ_\mu$. Consider decreasing $\succ_\mu$-chains starting from a term t. If $s \to^*_{U_{cs}(\mathcal{R})} t$ for some term $s \in \mathcal{T}(\Sigma, V)$ (i.e., t is reachable from an original term), there cannot be an infinite decreasing $\succ_\mu$-chain starting from t because this would contradict termination of $\to_{U_{cs}(\mathcal{R})}$ on $\mathcal{T}(\Sigma, V)$. Otherwise, $t = C[t_1 \ldots t_n]_{p_1 \ldots p_n}$, such that $s_i \to^*_{U_{cs}(\mathcal{R})} t_i$, $s_i \in \mathcal{T}(\Sigma, V)$ and $p_i \in Pos^\mu(t)$ for all $i \in \{1, \ldots, n\}$ and the same is true for no proper superterm of any t_i. Thus, if $t \succ_\mu u$, then $u = C[t_1 \ldots u_i \ldots t_n]_{p_1 \ldots p_i \ldots p_n}$ and $t_i \succ u_i$. Furthermore, if $u \succ_\mu v$, then $v = C[t_1 \ldots u_i \ldots v_j \ldots t_n]_{p_1 \ldots p_i \ldots p_j \ldots p_n}$ and $t_j \succ v_j$. It is easy to see that there cannot be an infinite decreasing $\succ_\mu$-sequence of this shape, as each decreasing $\succ$-sequence starting at some t_i is finite. Hence, $\succ_\mu$ is well-founded.

If we have $\sigma s_i \succ_\mu \sigma t_i$ for all $1 \le i < j$, then we get (cf., the proof of Theorem 3) $\sigma l \to^*_{U_{cs}(\mathcal{R})} \sigma U(s_j, x_1, \ldots, x_m)$ and thus $\sigma l \succ^{st}_\mu \sigma s_i$ for all rules $l \to r \Leftarrow s_1 \to^* t_1, \ldots, s_n \to^* t_n$, all $0 \le j \le n$ and all substitutions $\sigma \colon V \to \mathcal{T}(\Sigma, V)$. Analogously, if $\sigma s_i \succeq_\mu \sigma t_i$ for all $1 \le i \le n$, then we have $\sigma l \to^*_{U_{cs}(\mathcal{R})} \sigma r$ and thus $\sigma l \succ_\mu \sigma r$.

Hence, $\mathcal{R}$ is cs-quasi-reductive. $\square$

As a corollary we obtain the following equivalences between the various notions.

Corollary 3. *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS. The following properties of $\mathcal{R}$ are equivalent: μ-termination of $U_{cs}(\mathcal{R})$ on original terms, cs-quasi-reductivity, quasi-decreasingness, and operational termination.*

5 Disproving Collapse-Extended Operational Termination

While proving termination on original terms is (at least theoretically) easier than proving general termination, disproving termination on original terms and thus disproving operational termination of DCTRSs might be significantly harder than ordinary non-termination analysis. However, in this section we show that the transformation of Definition 4 is complete with respect to collapse-extended termination (C_E-termination), thus solving an open problem from [12]. Hence, if a transformed system can be proved to be non-terminating, we can deduce non-C_E-operational termination of the underlying DCTRS.

Furthermore, whenever operational termination and C_E-operational termination of a DCTRS $\mathcal{R}$ coincide, then $U_{cs}(\mathcal{R})$ is μ-terminating if and only if $\mathcal{R}$ is operationally terminating.

Definition 7 (C_E-termination, [18, 29])**.** *We call a CSRS $\mathcal{R}$ with replacement map μ C_E-μ-terminating (or just C_E-terminating) if $\mathcal{R} \uplus \{G(x,y) \to x, G(x,y) \to y\}$*[5] *with $\mu(G) = \{1,2\}$ is μ-terminating.*

Definition 8 (C_E-cs-quasi-reductivity)**.** *Let $\mathcal{R}$ be a DCTRS. We call $\mathcal{R}$ C_E-cs-quasi-reductive if $\mathcal{R} \uplus \{G(x,y) \to x, G(x,y) \to y\}$ is cs-quasi-reductive.*

Lemma 2. *Let $U_{cs}(\mathcal{R}) = (U(\Sigma), U(R))$ be a CSRS obtained by the transformation of Definition 4 from a DCTRS $\mathcal{R} = (\Sigma, R)$. If $U_{cs}(\mathcal{R})$ is not μ-terminating, then there exists an infinite reduction chain starting from a term t, such that $root(t) \in \Sigma$ and every replacing subterm of t is μ-terminating.*

Proof. We denote by $U_1^\alpha, \ldots, U_n^\alpha$ the U-symbols introduced when transforming a conditional rule α (cf. Definition 2). Assume towards a contradiction that $U_{cs}(\mathcal{R})$ is not μ-terminating and no term t with the described properties exists. Thus, there exists a U-term u that is non-terminating where every proper replacing subterm of u is μ-terminating, because the existence of a not μ-terminating term containing only μ-terminating μ-replacing subterms is obvious and this term cannot have a root symbol from Σ because of our assumption. Hence, there exists an infinite reduction sequence D starting from u. We inspect D.

[5] We use the notation $\mathcal{R} \uplus \{G(x,y) \to x, G(x,y) \to y\}$ as abbreviation for $(\Sigma \uplus \{G\}, R \uplus \{G(x,y) \to x, G(x,y) \to y\})$.

Let $u = U_1^\alpha(s_1, u_1, \ldots, u_n)$. Since s_1 is terminating, u must reduce to a term $U_1^\alpha(s_1', u_1, \ldots, u_n)$ and further to $u' = U_2^\alpha(s_2, u_1, \ldots, u_n, t_1, \ldots, t_m)$ in D. The terms $t_1, \ldots, t_m$ occurred at replacing positions in s_1' and are thus terminating. Hence, according to our assumption either u' is still minimal non-terminating in the sense that all replacing subterms are terminating, or the new minimal non-terminating term is one (or a subterm) of the u_i that occurs now at a replacing position in s_2, because all other potential minimal non-terminating terms are rooted by original symbols contradicting our assumption of the non-existence of such terms.

In the case where u' is still minimal non-terminating, eventually another root step will be performed yielding u''. As before we argue that if after this step there exists a minimal non-terminating proper subterm, it must be a term u_i (or a subterm thereof) that was already present (non-replacing) in u.

Now note that there cannot be infinitely many such root reduction steps, since eventually (after finitely many root steps) u will be reduced to a term u^* with a root symbol from Σ. At this point we know from our assumption that there must be a non-terminating proper subterm of u^* (if there has not already been one before). Furthermore, we know that this subterm must be one (or a subterm) of the terms u_i that was already present in u.

Thus, from our assumption we deduce that the existence of a minimal non-terminating term u implies the existence of a minimal non-terminating term $u_i \lhd u$. Well-foundedness of $\lhd$ yields the desired contradiction. $\square$

Definition 9 (partial evaluation)**.** *Let $\mathcal{R} = (U(\Sigma), U(R))$ be a CTRS obtained from a DCTRS (Σ, R) by the transformation of Definition 4 and let t be a term such that every maximal U-rooted subterm of t is μ-terminating. Then we define $peval_{\mathcal{R}}(t)$ as*

$$peval_{\mathcal{R}}(t) = \begin{cases} x, & \text{if } t = x \in V \\ f(peval_{\mathcal{R}}(v_1), \ldots, peval_{\mathcal{R}}(v_n)), & \\ \qquad \text{if } t = f(v_1, \ldots, v_n) \text{ and } f \in \Sigma & \\ G'(peval_{\mathcal{R}}(u_1), \ldots, peval_{\mathcal{R}}(u_m)), & \\ \qquad \text{if } t = U_i^\alpha(v_1, \ldots, v_n) \text{ and } U_i^\alpha \notin \Sigma & \end{cases}$$

where $G'(g_1, \ldots, g_k)$ stands for $G(g_1, G(g_2, \ldots G(g_{k-1}, g_k) \ldots))$ and the terms u_i are the maximal (w.r.t. $\to_{\mathcal{R},\mu}$) terms such that $t \to^+_{\mathcal{R},\mu} u_i$ and $root(u_i) \in \Sigma$, in an arbitrary but fixed order. The symbol G is new and defined as non-deterministic projection symbol (i.e., $G(x, y) \to x$, $G(x, y) \to y$).

Observation 1. *An important property of peval*[6] *is that whenever $s \to_{U_{cs}(\mathcal{R})} t$, then $peval(s) \to^*_{U_{cs}(\mathcal{R})} peval(t)$. Moreover, obviously $s = t$ implies $peval(s) = peval(t)$.*

Informally, $peval(t)$ represents all maximal successors of t (w.r.t. $\to_{U_{cs}(\mathcal{R})}$) that do not contain any U-symbols. From this point of view, Observation 1 states the obvious fact that no new such successors are added in the transition from s to t if $s \to_{U_{cs}(\mathcal{R})} t$.

[6] The reference to $\mathcal{R}$ is omitted if it is clear from the context or of no relevance.

Theorem 6 (completeness for C_E-termination)**.** *Let $\mathcal{R}$ be a DCTRS and let $U_{cs}(\mathcal{R})$ its transformed system according to Definition 4. $\mathcal{R}$ is C_E-cs-quasi-reductive if and only if $U_{cs}(\mathcal{R})$ is C_E-terminating.*

Proof. $U_{cs}(\mathcal{R}^{C_E}) = U_{cs}(\mathcal{R}) \uplus \{G(x,y) \to x, G(x,y) \to y\}$ and $\mathcal{R}^{C_E} = \mathcal{R} \uplus \{G(x,y) \to x, G(x,y) \to y\}$. Note that $U_{cs}(\mathcal{R}^{C_E})$ is the system obtained by transforming $\mathcal{R}^{C_E}$.

The *if* part of the proof is therefore covered by Theorem 5, because termination of $U_{cs}(\mathcal{R}^{C_E})$ implies cs-quasi-reductivity of $\mathcal{R}^{C_E}$.

The *only if* part of the theorem will be proved indirectly by showing that non-termination of $U_{cs}(\mathcal{R}^{C_E})$ implies non-termination of $U_{cs}(\mathcal{R}^{C_E})$ on terms of the original signature of $\mathcal{R}$ (plus $\{G\}$), which further implies non-cs-quasi-reductivity of $\mathcal{R}^{C_E}$ according to Theorem 4.

So assume $U_{cs}(\mathcal{R}^{C_E})$ is non-terminating. According to Lemma 2 there exists an infinite reduction sequence (w.r.t. $U_{cs}(\mathcal{R}^{C_E})$) D starting from a term t_0 with a root symbol from $\Sigma \uplus \{G\}$, such that each replacing subterm of t_0 is terminating. We will prove the existence of another infinite reduction D' starting at $t'_0 = peval_{U_{cs}(\mathcal{R}^{C_E})}(t_0)$, which does not contain any U-symbols. Note that $t_0 = C[u_1 \dots u_n]_{p_1 \dots p_n}$ where the terms u_i are the maximal U-rooted subterms of t_0 and $t'_0 = C[peval(u_1) \dots peval(u_n)]_{p_1 \dots p_n}$. Moreover, u_i is terminating for all i. These two properties will serve as an invariant for all terms t_i and t'_i in D resp. the parallel reduction D'.

For notational simplicity, in the following we write $\to$ instead of $\to_{U_{cs}(\mathcal{R}^{C_E})}$. Consider the beginning of D. First, assume a reduction step $t_j \to t_{j+1}$ takes place at or below some position p_i ($t_j|_{p_i} = u_i \to \overline{u_i}$). According to Observation 1, $peval(u_i) \to^* peval(\overline{u_i})$ and thus $t_j[\overline{u_i}]_{p_1}$ and $t'_j[peval(\overline{u_i})]_{p_1}$ satisfy the invariant. Eventually, there must be a reduction step $t_k \to t_{k+1}$ at a position *above* one of the p_i, because all terms u_i were assumed to be terminating. Let $t_k|_q = \sigma l$ for some rule pattern l. In l there are no U-symbols except possibly at the root position. Thus, U-rooted subterms of t_k can only interfere with the matching through non-linearity of l. From Observation 1 we know that if two terms s and t are equal, then also $peval(s) = peval(t)$. Hence, $t'_k|_q = \sigma' l$ and furthermore t_{k+1} and t'_{k+1} satisfy the invariant. Thus, we have shown that the start of D, $t_0 \to^+ t_m$, corresponds to a non-empty reduction $t'_0 \to^+ t'_m$ where t_m and t'_m satisfy the given invariant. Hence, by induction the existence of an infinite reduction sequence D' issuing from t'_0 follows. □

As corollaries of Theorem 6 we get the following modularity results.

Corollary 4. *The property of C_E-cs-quasi-reductivity is modular for disjoint unions.*

Corollary 5. *C_E-operational termination (defined for a DCTRS $\mathcal{R}$ as operational termination of $\mathcal{R} \uplus \{G(x,y) \to x, G(x,y) \to y\}$) is modular for disjoint unions.*

Table 1 summarizes the relations between a DCTRS $\mathcal{R}$ and $U_{cs}(\mathcal{R})$.

Property of $U_{cs}(\mathcal{R})$	**Implied property of $\mathcal{R}$**	**Proved in**
Termination	Operational termination	Theorem 3 and Proposition 2
Non-termination	Non-(C_E-operational termination)	Theorem 6
Termination on original terms	Operational termination	Theorem 5 and Proposition 2
Non-(termination on original terms)	Non-(operational termination)	Theorem 4
C_E-termination	C_E-operational termination	Theorem 6
Non-(C_E-termination)	Non-(C_E-operational termination)	Theorem 6

Table 1: Properties of $U_{cs}(\mathcal{R})$ and the implied properties of a DCTRS $\mathcal{R}$.

6 Proving Termination on the Set of Original Terms

Theorem 5 suggests that in order to prove operational termination of a DCTRS $\mathcal{R}$, termination of $U_{cs}(\mathcal{R})$ on original terms has to be proved. However, although termination on original terms is a weaker property than ordinary termination, its analysis might be harder and has, despite being an interesting problem, to the authors' knowledge, rarely been investigated.

In the following, we introduce a simple approach to deal with this problem based on the dependency pair framework of [16]. We refer to the property of a CSRS $((\Sigma, R), \mu)$ being μ-terminating on a set of terms identified by a sub-signature Σ' of Σ as *(Σ')-sub-signature termination* or just sub-signature termination if Σ' is clear from the context.

In our setting we extend the notion of dependency pair problem, in order to take into account our intention of proving only termination on restricted sets of terms, by adding an additional value specifying a (sub-)signature. Thus, we define SS-CS-DP-problems (*sub-signature context-sensitive dependency pair problems*) to be *quadruples* $(DP, \mathcal{R}, \mu, \Sigma')$ where $DP = (\Sigma^\sharp, R^\sharp)$ and $\mathcal{R} = (\Sigma, R)$ are TRSs, μ is a replacement map for the combined signature $\Sigma^\sharp \cup \Sigma$, and $\Sigma' \subseteq \Sigma$ is a signature determining the starting terms, whose μ-termination we are interested in. An SS-CS-DP-problem $(DP, \mathcal{R}, \mu, \Sigma')$ is finite if there is no infinite $(DP, \mathcal{R}, \mu)$-chain starting with a dependency pair $u_1 \to v_1$ and using a substitution σ such that $\sigma u_1 \in \mathcal{T}((\Sigma^\sharp \setminus \Sigma) \cup \Sigma', V)$. Analogously to the case without subsignature restriction dealt with in [1, Theorem 12], we can characterize termination of a CSRS on terms identified by a subsignature by finiteness of a corresponding SS-CS-DP-problem.

Proposition 4. *A TRS $\mathcal{R} = (\Sigma, R)$ with replacement map μ is μ-terminating on terms $\mathcal{T}(\Sigma', R)$ if and only if the SS-CS-DP-problem $(DP(\mathcal{R}, \mu), \mathcal{R}, \mu, \Sigma')$ is finite.*

Following the dependency pair framework of [16], an *SS-CS-dependency pair processor* (SS-CS-DP-processor) is a function *Proc* that takes as input an SS-CS-DP-problem and returns either a set of SS-CS-dependency pair problems or "`no`". We call an SS-CS-DP-processor *sound* if finiteness of all SS-CS-DP-problems in $Proc(d)$ implies finiteness of the input SS-CS-DP-problem d. An SS-CS-DP-processor is *complete* if for all SS-CS-DP-problems d, d is infinite whenever $Proc(d)$ is "`no`" or $Proc(d)$ contains an infinite SS-CS-DP-problem.

We introduce two SS-CS-DP-processors that are tailored to the task of proving finiteness of SS-CS-DP-problems. These processors build upon the well-known narrowing processor for the dependency pair framework (see e.g. [16]).

The basic idea of this processor is to anticipate the first step of all possible rewrite sequences in a potential dependency pair chain between two dependency pairs. If $\sigma v_i \to^* \sigma u_{i+1}$ is part of a chain and σv_i and σu_{i+1} are not equal (actually we demand that v_i and u_{i+1} are not unifiable) then the rewrite sequence $\sigma v_i \to^* \sigma u_{i+1}$ is non-empty and contains at least one reduction step at a position $p \in Pos_\Sigma(v_i)$ (see the proof of Theorem 7 for a justification of this claim). Thus, all possibilities of the first such step are covered by replacing $u_i \to v_i$ by the set $\{\theta_j u_i \to v_i^j \mid 1 \leq j \leq n\}$ with $v_i^1, \dots v_i^n$ being all possible (one step, context-sensitive) narrowings of v_i and $\theta_1, \dots, \theta_n$ being the corresponding mgu's. Theorem 7 below shows that replacing a rule $u_i \to v_i \in DP$ in an SS-CS-DP-problem $\mathcal{P} = (DP, \mathcal{R}, \mu, \Sigma')$ by the set of narrowings does neither alter finiteness nor infinity of $\mathcal{P}$ provided that v_i is linear and does not unify with a left-hand side of any rule in DP.

Analogously, a rule $u_i \to v_i$ occurring in a chain can be replaced under the corresponding preconditions by the set $\{u_i^j \to \theta_j v_i \mid 1 \leq j \leq m\}$, where $u_i^1, \dots u_i^m$ are the (one step, context-sensitive) *backward* narrowings of u_i and $\theta_1, \dots, \theta_m$ are the corresponding mgu's.

Applying these narrowing approaches in proofs of termination of CSRSs, obtained from DCTRSs by the transformation of Definition 4, allows us to restrict the set of narrowings that we have to consider.

The following lemmata provide the basis for this restriction. Lemma 3 states that the evaluation of conditions inside U-terms is only necessary if the U-term can eventually be reduced to an original term, i.e., if the conditions are satisfiable. Lemma 4 states that in a chain whose initial term does not contain U-symbols no U-terms can occur that are not reachable by an original term.

Lemma 3. *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS. Assume that $u_1 \xrightarrow{\epsilon} v_1 \xrightarrow{>\epsilon}{}^* u_2 \xrightarrow{\epsilon} v_2 \dots$ is an infinite $(DP(U_{cs}(\mathcal{R})), U_{cs}(\mathcal{R}), \mu)$-chain and let σ be a substitution enabling this chain. If the term σu_1 does not contain any U-symbol, then there also exists an infinite $(DP(U_{cs}(\mathcal{R})), U_{cs}(\mathcal{R}), \mu)$-chain, such that for each term $f^\sharp(t_1, ..., t_n)$ in this chain, each subterm t_i is* reducible *to a term from $\mathcal{T}(\Sigma, V)$.*

Lemma 4. *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS. Assume that $u_1 \xrightarrow{\epsilon} v_1 \xrightarrow{>\epsilon}{}^* u_2 \xrightarrow{\epsilon} v_2 \dots$ is an infinite $(DP(U_{cs}(\mathcal{R})), U_{cs}(\mathcal{R}), \mu)$-chain and let σ be a substitution enabling this chain. If the term σu_1 does not contain any U-symbol, then no term in this chain contains a U-term that is not* reachable *by a term from $\mathcal{T}(\Sigma, V)$.*

Lemmata 3 and 4 motivate the definition of two dependency pair processors based on the standard narrowing processor.

Definition 10 (restricted forward narrowing). *Let $(DP, \mathcal{R}, \mu, \Sigma')$ be an SS-CS-DP-problem with $\mathcal{R} = (\Sigma, R)$. If $u_i \to v_i \in DP$, $\overline{Var^\mu}(u_i) \cap Var^\mu(v_i) = \emptyset$, v_i is not unifiable with any left-hand side of a rule in DP and v_i is linear, then $Proc_{\text{RFN}}$ yields a new SS-CS-DP-problem $(DP', \mathcal{R}, \mu, \Sigma')$ where*

$$DP' = (DP - \{u_i \to v_i\}) \cup \{\theta_k u_i^k \to v_i^k \mid 1 \leq k \leq n\}$$

and $v_i^1, \ldots, v_i^n$ are the (one-step, context-sensitive) narrowings of v_i with corresponding mgu's $\theta_1, \ldots, \theta_n$, such that the v_i^k contain only those Σ-terms that can be instantiated in a way such that they are reducible *to Σ'-terms for all $1 \leq k \leq n$.*

Theorem 7. *The dependency pair processor $Proc_{\text{RFN}}$ is sound and complete for an SS-CS-DP-problem $(DP, \mathcal{R}, \mu, \Sigma')$ where $DP = (\Sigma^\sharp, R^\sharp)$ and $\mathcal{R} = (\Sigma, R)$ provided that $\mathcal{R} = U_{cs}(\mathcal{R}')$ for some DCTRS $\mathcal{R}'$ and $\Sigma^\sharp \cap (\Sigma \setminus \Sigma') = \emptyset$ (i.e., $\Sigma^\sharp$ does not contain any U-symbols).*

Proof. Soundness: Let $P = (\mathcal{P}, \mathcal{R}, \mu, \Sigma')$ be the initial SS-CS-DP-problem. Lemma 3 shows that if P is infinite then there exists an infinite dependency pair chain containing only such U-terms that are reducible to Σ'-terms. Let $v_1 \to u_1, \ldots, v_i \to u_i, s \to t, v_{i+1} \to u_{i+1}, \ldots$ be such a chain. Thus, there is a substitution σ with $\sigma u_j \to^*_{\mathcal{R},\mu} \sigma v_{j+1}$ for all $\{j > 0 \mid j \neq i\}$, $\sigma u_i \to^*_{\mathcal{R},\mu} \sigma s$, $\sigma t \to^*_{\mathcal{R},\mu} \sigma v_{i+1}$ and such that the reduction sequence $\sigma t \to^*_{\mathcal{R},\mu} \sigma v_{i+1}$ has minimal length (among all possible substitutions).

We take a closer look at the sequence $\sigma t \to^*_{\mathcal{R},\mu} \sigma v_{i+1}$ and show that due to the minimality of its length the first reduction step must take place at a position $p \in Pos_\Sigma(t)$: Assume that the first step is at position $q \notin Pos_\Sigma(t)$ and $t|_q = x$. Thus

$$\sigma t \xrightarrow{q} t' = \sigma' t \to^* \sigma v_{i+1}$$

We define a new substitution σ' by $\sigma' x = t'|_q$ and $\sigma' y = \sigma y$ for all $y \neq x$. Since all pairs on a chain are considered to be variable disjoint, we have $\sigma' u_i = \sigma u_i \to^*_{\mathcal{R},\mu} \sigma s \to_{\mathcal{R},\mu} \sigma' s$, $\sigma' t \to^*_{\mathcal{R},\mu} \sigma' v_{i+1}$ and $\sigma' v_j \to^*_{\mathcal{R},\mu} \sigma' u_{j+1}$ for all $\{j > 0 \mid j \neq i\}$. Thus, the reduction sequence $\sigma' t \to^*_{\mathcal{R},\mu} \sigma' v_{i+1}$ has a smaller length than $\sigma t \to^*_{\mathcal{R},\mu} \sigma v_{i+1}$ which contradicts our minimality assumption for σ. Note that the existence of the subsequence $\sigma s \to_{\mathcal{R},\mu} \sigma' s$ is guaranteed by the fact that $\overline{Var^\mu}(s) \cap Var^\mu(t) = \emptyset$.

Hence, the sequence $\sigma t \to^*_{\mathcal{R},\mu} \sigma v_{i+1}$ starts with a reduction step at position $p \in Pos_\Sigma(t)$. We assume that the reduction sequence is non-empty, otherwise t and v_{i+1} would unify. Moreover, t is assumed to be linear. We show that there is a narrowing $\bar{t}$ of t obtained by narrowing t with mgu θ, such that $v_1 \to u_1, \ldots v_i \to u_i, \theta s \to \bar{t}, v_{i+1} \to u_{i+1}, \ldots$ is an infinite chain and each term in this chain can be instantiated such that it can be reduced to a Σ'-term.

The reduction sequence $\sigma t \to^*_{\mathcal{R},\mu} \sigma v_{i+1}$ starts with a single reduction $\sigma t = \sigma t[\rho l]_p \to_{\mathcal{R},\mu} \sigma t[\rho r]_p$ using a rule $l \to r$. Since we consider l and t to be variable disjoint, we extend σ so that $\sigma x = \rho x$ for all $x \in Dom(\rho)$. Thus, σ unifies l and $t|_p$ and there is also an mgu θ for l and $t|_p$ ($\sigma = \tau \circ \theta$).

Then t narrows to $\bar{t} = t[\theta r]_p$ and since $\theta s \to \bar{t}$ is assumed to be variable disjoint from all other pairs in a chain, we can adapt σ to behave like τ on the variables of θs and $\bar{t}$. Thus,

$$\begin{aligned} \sigma u_i \to^*_{\mathcal{R},\mu} \sigma s &= \tau\theta s = \sigma\theta s \\ \sigma\bar{t} = \tau\bar{t} = \tau\theta t[\tau\theta r]_p &= \sigma t[\sigma r]_p = \sigma t[\rho r]_p \to^*_{\mathcal{R},\mu} \sigma v_{i+1} \end{aligned}$$

and $v_1 \to u_1, \dots v_i \to u_i, \theta s \to \bar{t}, v_{i+1} \to u_{i+1}, \dots$ is an infinite chain. Moreover, an instance (obtained through σ) of each subterm of $\bar{t}$ is reducible to a Σ'-term, because this was true for the chain we started with and all terms of the new chain occurred already in the original one. Thus, we showed that infinity of an SS-CS-DP-problem P implies infinity of the problem $Proc_{\text{RFN}}(P)$.

Completeness: Let $P = (\mathcal{P} \cup \{s \to t\}, \mathcal{R}, \mu, \Sigma')$ be an SS-CS-DP-problem such that t is linear and does not unify with any left-hand side of a rule in $\mathcal{P}$, and let $(\mathcal{P} \cup \{\theta_1 s \to t_1, \dots \theta_n s \to t_n\}, \mathcal{R}, \mu, \Sigma')$ be $Proc_{\mathtt{rfn}}(P)$. We show that if $v_1 \to u_1, \dots, v_i \to u_i, \theta_m s \to t_m, v_{i+1} \to u_{i+1}, \dots$ is a $(\mathcal{P} \cup \{\theta_1 s \to t_1, \dots \theta_n s \to t_n\}, \mathcal{R}, \mu)$-chain for some $1 \le m \le n$, then $v_1 \to u_1, \dots, v_i \to u_i, s \to t, v_{i+1} \to u_{i+1}, \dots$ is a chain as well.

As $v_1 \to u_1, \dots, v_i \to u_i, \theta_j s \to t_j, v_{i+1} \to u_{i+1}, \dots$ is a chain, there is substitution a σ such that $\sigma u_j \to^*_{\mathcal{R},\mu} \sigma v_{i+1}$ for all $\{j > 0 \mid j \neq i\}$, $\sigma u_i \to^*_{\mathcal{R},\mu} \sigma\theta_m s$ and $\sigma t_m \to^*_{\mathcal{R},\mu} \sigma v_{i+1}$.

As $s \to t$ does not share any variables with the rules $v_j \to u_j$ for all $j > 0$, we can modify σ to behave like $\sigma\theta$ on the variables of $s \to t$. Thus, we have

$$\sigma u_i \to^*_{\mathcal{R},\mu} \sigma\theta s = \sigma s$$

and because of $\theta t \to_{\mathcal{R},\mu} t_m$ (by the definition of context-sensitive narrowing) we get

$$\sigma t = \sigma\theta t \to^*_{\mathcal{R},\mu} \sigma t_m \to^*_{\mathcal{R},\mu} \sigma v_{i+1}$$

Thus, $v_1 \to u_1, \dots, v_i \to u_i, s \to t, v_{i+1} \to u_{i+1}, \dots$ is a chain and we can construct a $(\mathcal{P} \cup \{s \to t\}, \mathcal{R}, \mu)$-chain out of a $(\mathcal{P} \cup \{\theta_1 s \to t_1, \dots \theta_n s \to t_n\}, \mathcal{R}, \mu)$-chain this way. $\square$

Note that the precondition of the narrowed dependency pair not containing variables that are forbidden in its left-hand side but allowed in its right-hand side is crucial as the following example illustrates.

Example 7. *Consider the following CSRS* $(\mathcal{R}, \mu)$

$$\begin{aligned} t(f(x)) &\to t(h(x)) \\ t(i(b)) &\to t(f(a)) \\ a &\to b \\ h(x) &\to i(x) \end{aligned}$$

with $\mu(t) = \mu(h) = \{1\}$ *and* $\mu(f) = \mu(i) = \emptyset$*. It is non-terminating as* $t(f(a)) \to^{+}_{\mathcal{R},\mu} t(f(a))$*. The dependency pairs* $DP(\mathcal{R})$ *are*

$$\begin{array}{rcl} t^{\#}(f(x)) & \to & t^{\#}(h(x)) \\ t^{\#}(i(b)) & \to & t^{\#}(f(a)) \\ t^{\#}(f(x)) & \to & h^{\#}(x) \\ t^{\#}(f(x)) & \to & D^{\#}(x) \\ D^{\#}(a) & \to & a^{\#} \end{array}$$

forming only one potential cycle consisting of the pairs

$$\begin{array}{rl} t^{\#}(f(x)) & \to t^{\#}(h(x)) \\ t^{\#}(i(b)) & \to t^{\#}(f(a)) \end{array}$$

The right-hand side of the first pair is linear and it does not unify with a left-hand side of any other pair. However, there are forbidden variables in its left-hand side that occur replacing in the right-hand side. These two dependency pairs form an infinite chain with $\mathcal{R}$ *as* $h(a) \to^{+}_{\mathcal{R},\mu} i(b)$*. However, if the right hand side of the first pair were narrowed to* $t^{\#}(i(x))$ *and the pair were replaced by* $t^{\#}(f(x)) \to t^{\#}(i(x))$*, then the infinite chain would no longer be possible, because* $i(a) \not\to_{\mathcal{R},\mu} i(b)$*. Thus, the application of the narrowing processor would be incorrect in this case.*

The second dependency pair processor makes use of backward narrowing.

Definition 11 (restricted backward narrowing)**.** *Let* $(DP, \mathcal{R}, \mu, \Sigma')$ *be an SS-CS-DP-problem with* $\mathcal{R} = (\Sigma, R)$*. If* $u_i \to v_i \in DP$, $\overline{Var^{\mu}}(v_i) \cap Var^{\mu}(u_i) = \emptyset$, u_i *is not unifiable with any right-hand side of a rule in* DP *and* u_i *is linear, then* $Proc_{\text{RBN}}$ *yields a new SS-CS-DP-problem* $(DP', \mathcal{R}, \mu, \Sigma')$ *where*

$$DP' = (DP - \{u_i \to v_i\}) \cup \{u_i^k \to \theta_k v_i^k \mid 1 \leq k \leq n\}$$

and $u_i^1, \ldots, u_i^n$ *are the (one-step, context-sensitive)* backward *narrowings of* u_i *with corresponding mgu's* $\theta_1, \ldots, \theta_n$*, such that* u_i^k *contains only* Σ*-terms that can be instantiated in a way so that are* reachable *from* Σ'*-terms for all* $1 \leq k \leq n$*.*

Theorem 8. *The dependency pair processor* $Proc_{\text{BFN}}$ *is sound and complete for an SS-CS-DP-problem* $(DP, \mathcal{R}, \mu, \Sigma')$ *where* $DP = (\Sigma^{\sharp}, R^{\sharp})$ *and* $\mathcal{R} = (\Sigma, R)$ *provided that* $\mathcal{R} = U_{cs}(\mathcal{R}')$ *for some DCTRS* $\mathcal{R}'$ *and* $\Sigma^{\sharp} \cap (\Sigma \setminus \Sigma') = \emptyset$ *(i.e.,* $\Sigma^{\sharp}$ *does not contain any* U*-symbols).*

Proof. Analogous to the proof of Theorem 7. □

The narrowing processors use the notions *reducible to* respectively *reachable from* which are both undecidable in general. Even worse, we are interested in reachability resp. reducibility of all possible instances of terms. Thus, in order to apply these processors in practice, we need to use heuristics to approximate these notions. A very simple approach would be to discard only those narrowings that are U-terms and (forward resp. backward) narrowing normal forms. Examples 8 and 9 below show that this simple approximation is already sufficient to prove termination on original terms where ordinary termination does not hold (Example 8), or to significantly reduce the number of narrowings that have to be considered (Example 9).

Apart from such simple approximations one could also think of more sophisticated ones. For instance in the "forward" approach non-reducibility to original terms could be detected by *root-stability* which is still undecidable but for which non-trivial decidable approximations exist (e.g. strong root stability [19]).

Example 8. *Consider the transformed CSRS $\mathcal{R}$ of Example 5 and the SS-CS-DP-problem $\mathcal{P}_0 = (DP_0, \mathcal{R}, \mu, \Sigma')$ where $DP_0 = DP(\mathcal{R})$, μ has been extended to take dependency pair symbols into account and Σ' is Σ minus all U-symbols. $DP(\mathcal{R}) = \{f^\sharp(k(a), k(b), x) \to f^\sharp(x, x, x)\}$[7] . Applying $Proc_{\text{RBN}}$ to $\mathcal{P}_0$, we obtain a new problem $\mathcal{P}_1 = (DP_1, \mathcal{R}, \mu, \Sigma')$ where*

$$\begin{aligned} DP_1 = \{f^\sharp(U_2(c(a), z), k(b), x) &\to f^\sharp(x, x, x),\\ f^\sharp(k(a), U_2(c(b), z), x) &\to f^\sharp(x, x, x)\}. \end{aligned}$$

$Proc_{\text{RBN}}$ can be applied again using either rule in DP_1 for narrowing. After iterated applications of $Proc_{\text{RBN}}$, all narrowings of left-hand sides of rules in DP_i contain the term $U_1(d, d)$ as their first or second argument. As this term is a backward narrowing normal form, $DP_{i+1} = \emptyset$ and we conclude termination on original terms according to Theorem 8.

Example 9. *Consider the transformed CSRS $\mathcal{R}$ of Example 2. We use forward narrowing on the rule.*

$$A^\sharp \to h^\sharp(f(a), f(b))$$

Thus, the pair is replaced by two new rules

$$\begin{aligned} A^\sharp &\to h^\sharp(U(a, a), f(b))\\ A^\sharp &\to h^\sharp(f(a), U(b, b)) \end{aligned}$$

$Proc_{\text{RFN}}$ can be applied again to the resulting problem, such that the right-hand sides of the new rules are narrowed. Eventually, one of the arguments of $h^\sharp$ will narrow to instances of $U(d, x)$, $U(k, x)$, $U(l, x)$ or $U(m, x)$. As all instances of these terms are root stable*, those narrowings can be disregarded according to Definition 10. Thus, in the row of SS-CS-dependency pair problems obtained by repeated application of $Proc_{\text{RFN}}$, the size of the TRSs (to be precise of the TRS in the first component of the tupels) will not grow as fast as it would, if no narrowings were discarded and smaller problems are obviously easier to handle (also with other dependency pair processors) than bigger ones. Indeed, termination of the CSRS of this example can be shown automatically with the described method (cf. Example 11 below).*

[7] Here, we restrict the set of dependency pairs to those that are possibly part of a cycle in the *dependency graph*. See [1, 2] for a motivation and justification of this approach.

In a sense, the transformation of Definition 4 distributes the evaluation of the conditions of one conditional rule among several unconditional rules. The results of these single evaluations are propagated through the variables from one unconditional rule to the next one. With our narrowing approach we try to approximate the results of single evaluations, but we still we need a way to propagate these results in proofs of termination.

To this end we propose an *instantiation* processor, whose informal goal it is to propagate the results of condition evaluations approximated through narrowing to subsequent conditions (i.e. subsequent rules in the transformed system).[8] The following lemma provides the theoretical basis for our instantiation processor.

Lemma 5. *Let $\mathcal{P} = (\Sigma, R)$ and $\mathcal{R} = (\Sigma, R')$ ($\Sigma = \mathcal{F} \uplus \mathcal{C}$) be TRSs with a combined replacement map μ. If $\theta s \xrightarrow{\epsilon}_{\mathcal{P},\mu} \theta t \xrightarrow{>\epsilon^*}_{\mathcal{R},\mu} \theta' s' \xrightarrow{\epsilon}_{\mathcal{P},\mu} \theta' t'$, $\sigma s' = t$ for some substitution σ, $\overline{Var^{\mu}}(t')$[9] $\cap Var^{\mu}(s') = \emptyset$ and all variables of s' are contained only in constructor subterms (w.r.t. $\mathcal{R}$) (i.e. $s'|_p \in Var \Rightarrow \forall q < p\colon s'|_q \in (\Sigma \cup \mathcal{C}) \setminus \mathcal{F}$), then $\theta\sigma s' \xrightarrow{\epsilon}_{\mathcal{P},\mu} \overline{\theta}\sigma t' \to^*_{\mathcal{R},\mu} \theta' t'$ for some $\overline{\theta}$, such that $\overline{\theta}x = \theta x$ for all $x \in Var(t)$.*

Definition 12 (backward instantiation processor)**.** *Let $(DP = \{s \to t\} \cup DP', \mathcal{R}, \mu, \Sigma')$ be an SS-CS-DP-problem with $\mathcal{R} = (\Sigma, R)$, such that all variables of s are contained only in constructor subterms of s (w.r.t. $\mathcal{R}$) and $\overline{Var}_{\mu}(t) \cap Var^{\mu}(s) = \emptyset$. The set $Pred_{s\to t} = \{l \to r \in DP \mid \gamma = mgu(cap(ren(r)), cap(ren(s))\}$ defines all potential predecessors of the pair $s \to t$ on $(DP, \mathcal{R}, \mu)$-chains. If, for all $l \to r \in Pred_{s\to t}$,$r = \sigma s$ for some σ, then the processor $Proc_{\text{BI}}$ yields $(DP' \cup \{\sigma s \to \sigma t \mid l \to r \in Pred_{s\to t} \wedge r = \sigma s\}, \mathcal{R}, \mu, \Sigma')$.*

Theorem 9. *The processor $Proc_{\text{BI}}$ is sound and complete.*

Proof. Soundness: Assume there is an infinite dependency pair chain w.r.t. to a DP problem $\mathcal{P} = (DP, \mathcal{R}, \mu, \Sigma)$. We show that there also exists an infinite chain w.r.t. to the problem $Proc_{\text{BI}}(\mathcal{P}) = \mathcal{P}'$.

Consider an arbitrary fragment of the initial infinite chain:

$$\ldots \theta t_i \to^*_{\mathcal{R},\mu} \theta' s_{i+1} \xrightarrow{\epsilon}_{DP} \theta' t_{i+1} \ldots .$$

Then, we can construct an analogous chain fragment in $Proc_{\text{BI}}(\mathcal{P})$, as either $s_{i+1} \to t_{i+1}$ is contained in the dependency pairs of the derived problem $\mathcal{P}'$, or $t_i = \sigma s_{i+1}$ and thus there is a dependency pair $\sigma s_{i+1} \to \sigma t_{i+1}$ in $\mathcal{P}'$. In the latter case the new chain fragment is

$$\ldots \theta t_i = \theta\sigma s_{i+1} \xrightarrow{\epsilon}_{\mathcal{P}'} \theta\sigma t_{i+1} \to^*_{\mathcal{R},\mu} \theta' t_{i+1}$$

(according to Lemma 5).

Completeness: Consider an infinite chain w.r.t. $\mathcal{P}'$. $\ldots \theta\sigma s_i \xrightarrow{\epsilon} \theta\sigma t_i \ldots$. As we assume that all dependency pairs in chains are variable disjoint we can adapt θ to behave like $\theta\sigma$ and thus obtain an infinite DP chain w.r.t. to the original problem $\mathcal{P}$. □

[8] Note that our instantiation processor is similar to, but incomparable with the one in [16]

[9] i.e., those variables of t' that occur at non-replacing positions.

Example 10. *Consider an SS-CS-DP-problem* $P = (DP, \mathcal{R}, \mu, \Sigma')$ *where*

$$DP \quad = \quad \begin{cases} d^\# & \rightarrow \quad U_1^\#(c) \\ U_1^\#(x) & \rightarrow \quad U_2^\#(f(x)) \\ U_2^\#(d) & \rightarrow \quad d^\# \end{cases}$$

$$\mathcal{R} \quad = \quad \begin{cases} d & \rightarrow \quad U_1(c) \\ U_1(x) & \rightarrow \quad U_2(f(x)) \\ U_2(d) & \rightarrow \quad d \\ f(d) & \rightarrow \quad d \\ c & \rightarrow \quad b \end{cases}$$

, $\mu(U_1^\#) = \mu(U_2^\#) = \mu(U_1) = \mu(U_2) = \mu(f) = \{1\}$ *and* $\Sigma' = \{c, d, f, b\}$. *The problem originates from the dependency pair analysis of the DCTRS* $\mathcal{R}$*:*

$$\begin{array}{rcl} d & \rightarrow & d \Leftarrow c \rightarrow^* x, f(x) \rightarrow^* d \\ f(d) & \rightarrow & d \\ c & \rightarrow & b \end{array}$$

The backward narrowing processor can be applied to P*. The dependency pair* $s \rightarrow t$ *is* $U_1^\#(x) \rightarrow U_2^\#(f(x))$ *and its only potential predecessor is* $d^\# \rightarrow U_1^\#(c)$*. Since all functions in* s *above the variable* x *are constructors (i.e.* x *is contained in a constructor context in* s*) and the variable of* t *is replacing (i.e.* $\overline{Var}_\mu(t) = \emptyset$*), the additional preconditions for the application of the processor are satisfied. Thus, according to Definition 12 the result of the application of the processor is one new dependency pair problem* $(DP', \mathcal{R}, \mu, \Sigma')$ *where*

$$DP' \quad = \quad \begin{cases} d^\# & \rightarrow \quad U_1^\#(c) \\ U_1^\#(c) & \rightarrow \quad U_2^\#(f(c)) \\ U_2^\#(d) & \rightarrow \quad d^\# \end{cases}$$

Note that finiteness of this resulting SS-CS-DP-problem can be easily shown be repeated application of the forward narrowing processor of Definition 10.

Example 11. *Inside the dependency pair framework termination on original terms of* $U_{cs}(\mathcal{R})$ *and thus operational termination of* $\mathcal{R}$ *for the DCTRS* $\mathcal{R}$ *from Example 2 can be proved by repeated application of forward narrowing and backward instantiation. Our experiments showed that* μ*-termination of* $U_{cs}(\mathcal{R})$ *is hard to prove using other, standard techniques for termination analysis, thus the introduced dependency pair processors seem crucial for this particular example.*

Analogously to the backward instantiation processor we can also define a processor for forward instantiation.

Definition 13 (forward instantiation processor)**.** *Let* $(DP = \{s \rightarrow t\} \cup DP', \mathcal{R}, \mu, \Sigma')$ *be an SS-CS-DP-problem with* $\mathcal{R} = (\Sigma, R)$*, such that all variables of* t *are contained only in constructor subterms of* t *(w.r.t.* $\mathcal{R}$*) and* $Var^\mu(t) \cap \overline{Var}_\mu(s) =$

$\emptyset$. *The set* $Succ_{s\to t} = \{l \to r \in DP \mid \gamma = mgu(cap(ren(t)), cap(ren(l))\}$ *defines all potential successors of the pair* $s \to t$ *on* $(DP, \mathcal{R}, \mu)$*-chains. If, for all* $l \to r \in Succ_{s\to t}$, $l = \sigma t$ *for some* σ*, then the processor* $Proc_{\text{FI}}$ *yields* $(DP' \cup \{\sigma s \to \sigma t \mid l \to r \in Succ_{s\to t} \wedge l = \sigma t\}, \mathcal{R}, \mu, \Sigma')$.

In order to prove soundness and completeness we proceed as for the backward instantiation processor and show the following lemma that is dual to Lemma 5.

Lemma 6. *Let* $\mathcal{P} = (\Sigma, R)$ *and* $\mathcal{R} = (\mathcal{F} \uplus \mathcal{C}, R')$ *be TRSs with a combined replacement map* μ*. If* $\theta s \xrightarrow{\epsilon}_{\mathcal{P},\mu} \theta t \xrightarrow{>\epsilon^*}_{\mathcal{R},\mu} \theta' s' \xrightarrow{\epsilon}_{\mathcal{P},\mu} \theta' t'$, $\sigma t = s'$ *for some substitution* σ, $\overline{Var}_\mu(s) \cap Var^\mu(t) = \emptyset$ *and all variables of* t *are contained only in constructor subterms (w.r.t.* $\mathcal{R}$*) (i.e.* $t|_p \in Var \Rightarrow \forall q < p \colon t|_q \in (\Sigma \cup \mathcal{C}) \setminus \mathcal{F}$*), then* $\theta s \to^*_{\mathcal{R},\mu} \overline{\theta}\sigma s$ *for some* $\overline{\theta}$*, such that* $\overline{\theta} x = \theta' x$ *for all* $x \in Var(\sigma t)$.

Theorem 10. *The processor* $Proc_{\text{FI}}$ *is sound and complete.*

Proof. SOUNDNESS: Assume there is an infinite dependency pair chain w.r.t. to a DP problem $\mathcal{P} = (DP, \mathcal{R}, \mu, \Sigma)$. We show that there also exists an infinite chain w.r.t. to the problem $Proc_{\text{FI}}(\mathcal{P}) = \mathcal{P}'$.

Consider an arbitrary fragment of the initial infinite chain:

$$\ldots \theta s_i \xrightarrow{\epsilon}_{DP} \theta t_i \to^*_{\mathcal{R},\mu} \theta' s_{i+1} \ldots .$$

Then, we can construct an analogous chain fragment in $Proc_{\text{FI}}(\mathcal{P})$, as either $s_i \to t_i$ is contained in the dependency pairs of the derived problem $\mathcal{P}'$, or $s_{i+1} = \sigma t_i$ and thus there is a dependency pair $\sigma s_i \to \sigma t_i$ in $\mathcal{P}'$. In the latter case the new chain fragment is

$$\ldots \theta s_i \to^*_{\mathcal{R},\mu} \overline{\theta}\sigma s_i \xrightarrow{\epsilon}_{\mathcal{P}'} \overline{\theta}\sigma t_i = \theta' s_{i+1}$$

(according to Lemma 6).

COMPLETENESS: Consider an infinite chain w.r.t. $\mathcal{P}'$. $\ldots \theta\sigma s_i \xrightarrow{\epsilon} \theta\sigma t_i \ldots$. As we assume that all dependency pairs in chains are variable disjoint, we can adapt θ to behave like $\theta\sigma$ and thus obtain an infinite DP chain w.r.t. to the original problem $\mathcal{P}$. □

Note that the narrowing and instantiation approach is just one out of many methods to analyze dependency pair problems for their finiteness in the setting of ordinary termination analysis. However, regarding the structure of the systems that we analyze and using the fact that they were obtained from CTRSs, narrowing and instantiation seem to be an adequate tool in our special setting, because they are in some cases able to identify those instances of left-hand sides of rules for which the conditions of the corresponding CTRS are satisfiable.

Example 12. *In Example 8, after several narrowing steps the first TRS of the SS-CS-DP-problem is empty, thus the conditions of the conditional rule are unsatisfiable. Note that this DCTRS* $\mathcal{R}$ *is operationally terminating while* $U_{cs}(\mathcal{R})$ *is not* μ*-terminating. Hence, operational termination cannot be verified with standard ordering-based methods. Thus, again the presented narrowing processor is crucial for a successful automatic proof of operational termination.*

Taking into account that finding such instances or identifying instances for which the conditions are not satisfiable is potentially crucial for proving or disproving termination of (transformed) systems, narrowing and instantiation are important tools for this task. Moreover, our narrowing dependency pair processors allow us to reduce the number of narrowings generated and thus make the narrowing approach more efficient in practice.

6.1 Experimental Evaluation and Practical Issues

In order to evaluate the practical use of the context-sensitive unraveling as well as our approach to prove termination on restricted sets of terms, we implemented both the transformation and our proposed dependency pair processors in the tool VMTL ([30]).[10] The results and details of our tests can be found at the tool's homepage.[10] Out of 24 tested systems our implementation was able to prove operational termination of 19.

The examples were taken from the termination problem database (TPDB)[11] and from standard literature on conditional term rewriting.

In our experiments other termination tools supporting conditional rewrite systems scored worse on this set of examples. This shows that the introduced narrowing and instantiation processors exploiting the theoretical results of Section 4 significantly increase the power of VMTL.

On the negative side repeated application especially of narrowing processors can be expensive with respect to execution time (and space). Inside the dependency pair framework DP processors may be applied to DP-problems in an arbitrary order. Choosing and fixing such an order can significantly influence the power and efficiency of a termination tool. In our experiments, the narrowing and instantiation approach was only tried after other ordering-based methods to prove finiteness of DP-problems, which are more efficient, failed. This strategy turned out to be the most efficient and powerful one.

For more details about the system, its characteristics, features and benchmarks we refer to [30].

7 Related Work and Discussion

We analyzed the context-sensitive modification of the unraveling transformation of DCTRSs into TRSs ([24, 25, 28, 29]). This transformation plays a crucial role in several approaches for the termination analysis of current programming and specification languages (cf., [23, 12]). Moreover, conditions are inherent features of several functional programming languages. Hence, methods for the analysis of conditional systems are of utmost importance when it comes to verify such programs.

With our characterization of operational termination by termination of a CSRS on original terms, on the one hand we gain the opportunity to disprove

[10] `http://www.logic.at/vmtl/`
[11] `http://www.lri.fr/ marche/tpdb/`

operational termination (cf. also [14]). On the other hand, the task of proving termination on original terms is (at least) theoretically easier than proving general termination. This latter aspect of proving termination of rewrite systems not on all terms, but only on a subset of all terms, is an instance of a general interesting problem which has hardly been studied so far (of course, it also applies to other properties like confluence, having the *normal form property* etc.). Little seems to be known on questions of this type. In our case, clearly more research is necessary for exploiting the fact that termination only needs to be proved for certain terms, but not (necessarily) for all ones.

In Section 6 we introduced a simple approach to address the problem of proving termination on the set of original terms. Benchmarks performed with the termination tool VMTL indicate the practical relevance of our method. In particular, VMTL managed to prove operational termination automatically for several DCTRSs for which other existing termination tools, using more traditional approaches, fail. However, our approach should be understood as only a starting point for the task of analyzing restricted termination and leaves plenty of space for future improvements. We also conjecture that termination analysis on a restricted set of terms may be of interest in several areas where transformations are used. It is very common that transformations introduce new (auxiliary) functions that may give rise to spurious reduction chains. Restricting the attention to reductions starting from original terms may be more adequate in many situations.

In Section 5 we introduced the notion of C_E-operational termination and proved its modularity. We also showed that the context-sensitive version of the unraveling transformation is sound and complete for C_E-operational termination. This indicates that DCTRSs for which the operational termination and the C_E-operational termination behavior differ have a certain (Toyama-like) pathological structure as in the unconditional case.

In [27] and [26] the same transformation as in the current paper (with refinements) is used for the simulation of conditional rewriting rather than for termination analysis. We proved that the context-sensitive transformation is simulation sound and simulation complete in their sense. Other related works on context-sensitive transformationsof DCTRSs are [11, 12]. There, the setting is even more general, since context-sensitive conditional systems with membership equational theories (motivated by corresponding Maude specifications and programs) are dealt with. However, only sufficent conditions for, but no precise characterization of operational termination of original conditional systems are given.

To summarize we see three main contributions of this paper:

1. An exact characterization of operational termination of DCTRSs by termination of CSRSs on original terms.

2. The basis for proving non-(operational termination) of DCTRSs by means of proving non-(μ-termination) of CSRSs. Furthermore, it was shown that with the transformation of Definition 4 it is possible to characterize C_E-operational termination of a DCTRS by C_E-μ-termination of a CSRS.

3. Finally we provided two simple dependency pair processors (the narrowing processors) that are specialized for the task of analyzing the termination behaviour of CSRSs obtained by our transformation and showed that with their help operational termination of systems can be verified where other existing methods fail (cf. e.g. Example 8).

References

[1] B. Alarcón, F. Emmes, C. Fuhs, J. Giesl, R. Gutiérrez, S. Lucas, P. Scheider-Kamp and R. Thiemann. Improving context-sensitive dependency pairs. Technical Report: Aachener Informatik Berichte (AIB), 2008-13

[2] Alarcón, B., Emmes, F., Fuhs, C., Giesl, J., Gutiérrez, R., Lucas, S., Scheider-Kamp, P.,Thiemann, R.: Improving context-sensitive dependency pairs. In: I. Cervesato, H. Veith and A. Voronkov (eds.), Proc. 15th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning (LPAR'08), LNCS, vol. 5330, pp. 636–651. Springer (2008)

[3] B. Alarcón, R. Gutiérrez, J. Iborra, and S. Lucas. Proving termination of context-sensitive rewriting with MU-TERM. *Electr. Notes Theor. Comput. Sci.*, 188:105–115, 2007.

[4] B. Alarcón, R. Gutiérrez, and S. Lucas. Context-sensitive dependency pairs. In S. Arun-Kumar and N. Garg, eds., *Proc. 26th International Conference on Foundations of Software Technology and Theoretical Computer Science (FST&TCS'06)*, *LNCS* 4337, pp. 297–308, Kolkata, India, 2006. Springer-Verlag.

[5] S. Antoy, B. Brassel and M. Hanus. Conditional narrowing without conditions. In *Proc. 5th International ACM SIGPLAN Conference on Principles and Practice of Declarative Programming (PPDP'03), Uppsala, Sweden, August 27-29, 2003*, pp. 20–31, 2003, ACM Press.

[6] T. Arts and J. Giesl. Termination of term rewriting using dependency pairs. *Theoretical Computer Science*, 236(1–2):133–178, 2000.

[7] F. Baader and T. Nipkow. *Term rewriting and all that.* Cambridge University Press, New York, NY, USA, 1998.

[8] M. Bezem, J. Klop, and R. Vrijer, editors. *Term rewriting systems.* Cambridge Tracts in Theoretical Computer Science 55. Cambridge University Press, Mar. 2003.

[9] P. Borovanský, C. Kirchner, H. Kirchner, and P.-E. Moreau. Elan from a rewriting logic point of view. *Theoretical Computer Science*, 285(2):155–185, 2002.

[10] M. Clavel, F. Durán, S. Eker, P. Lincoln, N. Martí-Oliet, J. Meseguer, and C. Talcott. The Maude 2.0 system. In R. Nieuwenhuis, ed., *Proc. 14th International Conference on Rewriting Techniques and Applications (RTA'03), Valencia, Spain, June 9-11, 2003*, *LNCS* 2706, pp. 76–87. Springer-Verlag, June 2003.

[11] F. Durán, S. Lucas, J. Meseguer, and C. Marché. Proving termination of membership equational programs. In *Proc. ACM SIGPLAN Workshop on Partial Evaluation and Semantics-based Program Manipulation (PEPM'04), Verona, Italy, August 24-25, 2004*, pp. 147-158–308, 2004, ACM Press.

[12] F. Durán, S. Lucas, J. Meseguer, C. Marché, and X. Urbain. Proving operational termination of membership equational programs. *Higher-Order and Symbolic Computation*, 21:59–88, 2008.

[13] H. Ganzinger and U. Waldmann. Termination proofs of well-moded logic programs via conditional rewrite systems. In M. Rusinowitch and J.-L. Remy, eds., *Proc. 3rd international Workshop on Conditional Term Rewriting Systems (CTRS'92), Pont-à-Mousson, France, July 8-10, 1992*, *LNCS* 656, pp. 430–437. Springer, 1993.

[14] J. Giesl, R. Thiemann, and P. Schneider-Kamp. Proving and disproving termination of higher-order functions. In B. Gramlich, ed., *Proc. 5th International Workshop on Frontiers of Combining Systems (FroCos'05), Vienna, Austria, September 19-21, 2005*, *LNCS* 3717, pp, 216–231. Springer, 2005.

[15] J. Giesl, R. Thiemann, P. Schneider-Kamp, and S. Falke. Automated termination proofs with AProVE. In V. van Oostrom, ed., *Proc. 15th International Conference on Rewriting Techniques and Applications (RTA'04), Aachen, Germany, June 3-5, 2004*, *LNCS* 3091, pp. 210–220. Springer-Verlag, 2004.

[16] Jürgen Giesl, René Thiemann, Peter Schneider-Kamp and Stephan Falke. Mechanizing and improving dependency pairs. *Journal of Automated Reasoning*, 37(3):155–203, 2006.

[17] K. Gmeiner and B. Gramlich. Transformations of conditional rewrite systems revisited. In A. Corradini and U. Montanari, eds., *Recent Trends in Algebraic Development Techniques (WADT'08 – Selected Papers)*, *LNCS*, Springer-Verlag, to appear, 2009.

[18] B. Gramlich and S. Lucas. Modular termination of context-sensitive rewriting. In C. Kirchner, ed., *Proc. 4th International Conference on Principles and Practice of Declarative Programming (PPDP'02), Pittsburgh, PA, USA, Oct. 2002*, pp. 50–61, ACM Press, 2002.

[19] R. Kennaway. A conflict between call-by-need computation and parallelism. In N. Dershowitz and N. Lindenstrauss, eds., *Proc. 4th Int. Workshop on Conditional Term Rewriting Systems (CTRS'94), Berlin 1994*, *LNCS* 968, pp. 247–261. Springer, 1994.

[20] S. Lucas. Context-sensitive computations in functional and functional logic programs. *Journal of Functional and Logic Programming*, 1998(1), January 1998.

[21] S. Lucas, B. Alarcon, R. Gutierrez. Improving the context-sensitive dependency graph. *Electronic Notes in Theoretical Computer Science*, 188:91–103, 2007.

[22] S. Lucas, C. Marché, and J. Meseguer. Operational termination of conditional term rewriting systems. *Inf. Process. Lett.*, 95(4):446–453, 2005.

[23] S. Lucas and J. Meseguer. Operational termination of membership equational programs: The order-sorted way. In *Proc. 7th International Workshop on Rewriting Logic and its Applications (WRLA'08)*, Electronic Notes in Theoretical Computer Science, 2008, to appear.

[24] M. Marchiori. Unravelings and ultra-properties. In M. Hanus and M. Rodríguez-Artalejo, eds., *Proc. 5th Int. Conf. on Algebraic and Logic Programming (ALP'96), Aachen, Germany, September 25-27, 1996*, *LNCS* 1139, pp. 107–121. Springer, Sept. 1996.

[25] M. Marchiori. On deterministic conditional rewriting. Technical Report MIT LCS CSG Memo n.405, MIT, Cambridge, MA, USA, Oct. 1997. 27 pages.

[26] N. Nishida, T. Mizutani, and M. Sakai. Transformation for refining unraveled conditional term rewriting systems. *Electr. Notes Theor. Comput. Sci.*, 174(10):75–95, 2007.

[27] N. Nishida, M. Sakai, and T. Sakabe. Partial inversion of constructor term rewriting systems. In J. Giesl, ed., *Proc. 16th Int. Conf. on Rewriting Techniques and Applications (RTA'05), Nara, Japan, April 19-21, 2005*, *LNCS* 3467, pp. 264–278. Springer, 2005.

[28] E. Ohlebusch. Transforming conditional rewrite systems with extra variables into unconditional systems. In H. Ganzinger, D. McAllester, and A. Voronkov, eds., *Proc. 6th Int. Conf. on Logic Programming and Automated Reasoning (LPAR'99), Tbilisi, Georgia*, *LNCS* 1705, pp. 111–130. Springer, Sept. 1999.

[29] E. Ohlebusch. *Advanced topics in term rewriting.* Springer-Verlag, London, UK, 2002.

[30] F. Schernhammer and B. Gramlich. VMTL – A modular termination laboratory. In R.Treinen, ed., *Proc. 20th International Conference on Rewriting Techniques and Applications (RTA'09), Brasilia, Brazil*, *LNCS*. Springer, to appear, 2009.

[31] T.-F. Serbanuta and G. Rosu. Computationally equivalent elimination of conditions. In F. Pfenning, ed., *Proc. 17th International Conference on Rewriting Techniques and Applications (RTA'06)*, *LNCS* 4098, pp. 19–34. Springer, 2006.

Appendix: Missing Proofs

Proposition 1 *If a DCTRS $\mathcal{R}$ is quasi-reductive, then it is cs-quasi-reductive.*

Proof. The result is obvious, since if a DCTRS is quasi-reductive with respect to a signature extension Σ' and an ordering $\succ$, then it is cs-quasi-quasi-reductive w.r.t. the same signature extension and the same ordering and the replacement map μ with $\mu(f) = \{1, \ldots, ar(f)\}$ for all $f \in \Sigma'$. $\square$

Proposition 2 *If a DCTRS $\mathcal{R}$ is cs-quasi-reductive, then it is quasi-decreasing.*

Proof. Let $\mathcal{R}$ be cs-quasi-reductive w.r.t. the ordering $\succ_\mu$. First, we show that $\rightarrow_\mathcal{R} \subseteq \succ_\mu$: Assume $s \rightarrow_\mathcal{R} t$ ($s, t \in \mathcal{T}(\Sigma, V)$). We will use induction on the depth of the rewrite step in order to prove $s \succ_\mu t$. Assume the step $s \rightarrow_\mathcal{R} t$ has depth 1, i.e., an unconditional rule (or a rule with trivially satisfied conditions) is applied. In this case $s \succ_\mu t$ follows immediately from cs-quasi-reductivity of $\mathcal{R}$ and μ-monotonicity of $\succ_\mu$.

Next, assume the step $s \rightarrow_\mathcal{R} t$ has depth $d > 1$. Thus, a rule $l \rightarrow r \Leftarrow s_1 \rightarrow t_1, \ldots, s_n \rightarrow t_n$ is applied (i.e., $s|_p = \sigma l$). From the applicability of the conditional rule it follows that σ can be extended to σ' such that $\sigma' s_i \rightarrow^*_\mathcal{R} \sigma' t_i$ for all $1 \leq i \leq n$. Moreover, each reduction step in each of these reduction sequences has a depth smaller than d. Thus, the induction hypothesis and transitivity of $\succ_\mu$ yield $\sigma' s_i \succeq \sigma' t_i$ for all $1 \leq i \leq n$. Hence, by cs-quasi-reductivity we get $\sigma' l \succ_\mu \sigma' r$, and finally $s \succ_\mu t$ by μ-monotonicity of $\succ_\mu$.

Next we prove that $\mathcal{R}$ is quasi-decreasing with respect to the ordering $> := \succ_\mu^{st} |_{\mathcal{T}(\Sigma,V) \times \mathcal{T}(\Sigma,V)}$:

1. $\rightarrow_\mathcal{R} \subseteq >$: Follows immediately from $\rightarrow_\mathcal{R} \subseteq \succ_\mu \subseteq >$ if we restrict attention to terms of the original signature.

2. $> = >_{st}$: Assume there is a term s which is a proper subterm of a term $t \in \mathcal{T}(\Sigma, R)$ ($t = C[s]_p$), such that $t \not> s$. This implies $t \not\succ_\mu^{st} s$, which contradicts the definition of $\succ_\mu^{st}$ as p is a replacing position of t (because all positions in t are replacing).

3. For every rule $l \rightarrow r \Leftarrow s_1 \rightarrow t_1, \ldots, s_n \rightarrow t_n$, every substitution $\sigma \colon V \rightarrow \mathcal{T}(\Sigma, V)$ and every $i \in \{0, \ldots, n-1\}$ we must show $\sigma s_j \rightarrow^* \sigma t_j$ for every $j \in \{1, \ldots, i\}$ implies $\sigma l > \sigma s_{i+1}$. We know that $\sigma s_j \rightarrow^* \sigma t_j \Rightarrow \sigma s_j \succeq_\mu \sigma t_j$. Because of cs-quasi-reductivity this implies $\sigma l \succ_\mu^{st} \sigma s_{j+1}$ and thus $\sigma l > \sigma s_{j+1}$, since $\sigma l, \sigma s_{j+1} \in \mathcal{T}(\Sigma, V)$.

$\square$

Theorem 1 *Let $\mathcal{R}$ be a DCTRS ($\mathcal{R} = (\Sigma, R)$). For every $s, t \in \mathcal{T}(\Sigma, V)$ we have: If $s \rightarrow_\mathcal{R} t$, then $s \rightarrow^+_{U_{cs}(\mathcal{R})} t$.*

Proof. We use induction on the depth of the step $s \rightarrow_\mathcal{R} t$. If $s \rightarrow_\mathcal{R} t$ with a rule $l \rightarrow r$ (i.e., an unconditional rule), then $l \rightarrow r \in U_{cs}(\mathcal{R})$ and thus $s \rightarrow_{U_{cs}(\mathcal{R})} t$. Assume $s \rightarrow_\mathcal{R} t$ with a rule $\alpha : l \rightarrow r \Leftarrow s_1 \rightarrow^* t_1, \ldots, s_n \rightarrow^* t_n$. Then

$s = C[\sigma l]_p$ and $t = C[\sigma r]_p$. All rewrite sequences $\sigma s_i \to^*_{\mathcal{R}} \sigma t_i$ have lower depths than $\sigma l \to_{\mathcal{R}} \sigma r$, thus we can apply the induction hypothesis to obtain the following rewrite sequence in the transformed system:

$$\begin{array}{lll} C[\sigma l]_p & \to_{U_{cs}(\mathcal{R})} & C[U_1^\alpha(\sigma s_1, \sigma Var(l))]_p \\ & \to^*_{U_{cs}(\mathcal{R})} & C[U_1^\alpha(\sigma t_1, \sigma Var(l))]_p \\ & \to_{U_{cs}(\mathcal{R})} & C[U_2^\alpha(\sigma s_2, \sigma Var(l), \sigma \mathcal{E}Var(t_1))]_p \\ & \to^*_{U_{cs}(\mathcal{R})} & \cdots \\ & \to^*_{U_{cs}(\mathcal{R})} & C[U_n^\alpha(\sigma t_n, \sigma Var(l), \sigma \mathcal{E}Var(t_1), \ldots, \\ & & \quad \sigma \mathcal{E}Var(t_{n-1}))]_p \to_{U_{cs}(\mathcal{R})} C[\sigma r]_p = t \end{array}$$

□

Proposition 3 *Let $\mathcal{R}$ be a DCTRS. If $U(\mathcal{R})$ is terminating, then $U_{cs}(\mathcal{R})$ is μ-terminating.*

Proof. The result is immediate, since we have $\to_{U(\mathcal{R})} \supseteq \to_{U_{cs}(\mathcal{R})}$. □

Corollary 3 *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS. The following properties of $\mathcal{R}$ are equivalent: μ-termination of $U_{cs}(\mathcal{R})$ on original terms, cs-quasi-reductivity, quasi-decreasingness, and operational termination.*

Proof. The equivalence of quasi-decreasingness and operational termination was proved in [22]. Theorem 5, Lemma 2 and Theorem 4 show: $\mu_{U_{cs}(\mathcal{R})}$-termination of $U_{cs}(\mathcal{R})$ on $\mathcal{T}(\Sigma, V) \Rightarrow$ cs-quasi-reductivity of $\mathcal{R} \Rightarrow$ quasi-decreasingness of $\mathcal{R}$ $\Rightarrow \mu_{U_{cs}(\mathcal{R})}$-termination of $U_{cs}(\mathcal{R})$ on $\mathcal{T}(\Sigma, V)$. □

Corollary 4 *C_E-cs-quasi-reductivity is modular for disjoint unions.*

Proof. Let $\mathcal{R}$ and $\mathcal{S}$ be DCTRSs with disjoint signatures that are both C_E-cs-quasi-reductive. According to Theorem 6, $U_{cs}(\mathcal{R})$ and $U_{cs}(\mathcal{S})$ are C_E-μ-terminating. In [18], modularity of C_E-μ-termination is proved. Thus, $U_{cs}(\mathcal{R}) \uplus U_{cs}(\mathcal{S})$ is C_E-μ-terminating. As $U_{cs}(\mathcal{R}) \uplus U_{cs}(\mathcal{S}) = U_{cs}(\mathcal{R} \uplus \mathcal{S})$, $\mathcal{R} \uplus \mathcal{S}$ is C_E-cs-quasi-reductive. □

Proposition 4 *A TRS $\mathcal{R} = (\Sigma, R)$ with replacement map μ is μ-terminating on terms $\mathcal{T}(\Sigma', R)$ if and only if the SS-CS-DP-problem $DP(\mathcal{R}, \mu), \mathcal{R}, \mu, \Sigma'$ is finite.*

Proof. IF: Assume $\mathcal{R}$ is not μ-terminating on original terms. Then there exists a sequence of terms

$$t_1 \xrightarrow{>\epsilon *}_{\mathcal{R},\mu} t_1' \xrightarrow{\epsilon}_{\mathcal{R},\mu} s_1 \trianglerighteq_\mu t_2 \xrightarrow{>\epsilon *}_{\mathcal{R},\mu} t_2' \xrightarrow{\epsilon}_{\mathcal{R},\mu} s_2 \trianglerighteq_\mu t_3 \xrightarrow{>\epsilon *}_{\mathcal{R},\mu} t_3' \xrightarrow{\epsilon}_{\mathcal{R},\mu} \ldots$$

such that $t_1 \in \mathcal{T}(\Sigma', V)$ and t_i and t_i' are minimal not μ-terminating for all i, i.e., there is an infinite reduction sequence starting from t_i (resp. t_i') but all their proper replacing subterms are terminating.

According to the proof of [1, Theorem 12] there also exists a $(DP(\mathcal{R},\mu),\mathcal{R},\mu)$-chain ($DP(\mathcal{R},\mu) = (\Sigma^\sharp, R^\sharp)$) starting with the term $t_1^\sharp = \sigma l^\sharp$ for some rule $l^\sharp \to r^\sharp \in DP(\mathcal{R},\mu)$. Clearly, $t_1^\sharp \in \mathcal{T}((\Sigma^\sharp \setminus \Sigma) \cup \Sigma', V)$.

ONLY IF: In the completeness part of the proof of [1, Theorem 12], an infinite reduction sequence in $(\mathcal{R},\mu)$ is constructed out of an infinite $(DP(\mathcal{R},\mu),\mathcal{R},\mu)$-chain in a way such that if the chain starts with a rule $l^\sharp \to r^\sharp$ and σ enables the chain, the constructed reduction sequence starts with the term σl. If $\sigma l^\sharp \in \mathcal{T}((\Sigma^\sharp \setminus \Sigma) \cup \Sigma', V)$ then $\sigma l \in \mathcal{T}(\Sigma', V)$. Note that for each infinite chain we can find a suffix, such that the root symbol of the first rule in the chain is not $D^\sharp$ (cf. [1, Theorem 12]). It is easy to see that the starting term of such a maximal tail does not contain functions from $\Sigma \setminus \Sigma'$ if the starting term of the whole chain did not, because $\mu(D^\sharp) = \emptyset$ and the rules defining $D^\sharp$ in $DP(\mathcal{R},\mu)$ do not introduce such symbols. Moreover, note that the minimality of the chain, which we do not demand in the proposition, is not used in the proof of [1, Theorem 12]. □

Lemma 3 *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS. Assume that $u_1 \xrightarrow{\epsilon} v_1 \xrightarrow{>\epsilon^*} u_2 \xrightarrow{\epsilon} v_2 \ldots$ is an infinite $(DP(U_{cs}(\mathcal{R})), U_{cs}(\mathcal{R}), \mu)$-chain and let σ be a substitution enabling this chain. If the term σu_1 does not contain any U-symbol, then there also exists an infinite $(DP(U_{cs}(\mathcal{R})), U_{cs}(\mathcal{R}), \mu)$-chain, such that for each term $f^\sharp(t_1, ..., t_n)$ in this chain, each subterm t_i is* reducible *to a term from $\mathcal{T}(\Sigma, V)$.*

Proof. Consider an infinite $(DP(U_{cs}(\mathcal{R})), U_{cs}(\mathcal{R}), \mu)$-chain starting from σt_1, which does not contain a U-symbol. The dependency pairs do not introduce U-symbols (only $U^\sharp$-symbols), thus it suffices to show that each reduction sequence $\sigma v_i \to^* \sigma u_{i+1}$ between two dependency pairs $u_i \to v_i$ and $u_{i+1} \to v_{i+1}$ implies the existence of the reduction sequence $tb(\sigma v_i) \to^* tb(\sigma u_{i+1})$. The latter implication follows directly from the proof of Theorem 2.

Thus, if there still were U-terms not reducible to original terms in the chain, they would eventually be erased through a rule eliminating variables. Assume $tb(\sigma v_i) \to^* s \to s'$ is a subsequence of the infinite chain where such a U-term is eliminated in the transition from s to s'. As $tb(\sigma v_i)$ is an original term, according to the proof of Theorem 2 we have $tb(\sigma v_i) \to^* tb(s) \to tb(s')$ and thus the introduction of the problematic U-term could have been avoided in the first place. □

Lemma 4 *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS. Assume that $u_1 \xrightarrow{\epsilon} v_1 \xrightarrow{>\epsilon^*} u_2 \xrightarrow{\epsilon} v_2 \ldots$ is an infinite $(DP(U_{cs}(\mathcal{R})), U_{cs}(\mathcal{R}), \mu)$-chain and let σ be a substitution enabling this chain. If the term σu_1 does not contain any U-symbol, then no term in this chain contains a U-term that is not* reachable *by a term from $\mathcal{T}(\Sigma, V)$.*

Proof. Assume there were a U-term in the DP-chain, that is not reachable by an original term. Following the argumentation in the proof of Lemma 1, this U-term as well as each sub U-term could be traced backwards, to the point where the U-symbol had been introduced, because ultimately the chain started from an original term. Hence, the U-term would be reachable by an original term, contradicting our assumption. $\square$

Lemma 5 *Let $\mathcal{P} = (\Sigma, R)$ and $\mathcal{R} = (\mathcal{F} \uplus \mathcal{C}, R')$ be TRSs with a combined replacement map μ. If $\theta s \xrightarrow{\epsilon}_{\mathcal{P},\mu} \theta t \xrightarrow{>\epsilon}{}^{*}_{\mathcal{R},\mu} \theta' s' \xrightarrow{\epsilon}_{\mathcal{P},\mu} \theta' t'$, $\sigma s' = t$ for some substitution σ, $\overline{Var}_{\mu}(t') \cap Var^{\mu}(s') = \emptyset$ and all variables of s' are contained only in constructor subterms (w.r.t. $\mathcal{R}$) (i.e. $s'|_p \in Var \Rightarrow \forall q < p\colon s'|_q \in (\Sigma \cup \mathcal{C}) \setminus \mathcal{F}$), then $\theta\sigma s' \xrightarrow{\epsilon}_{\mathcal{P},\mu} \overline{\theta}\sigma t' \rightarrow^{*}_{\mathcal{R},\mu} \theta' t'$ for some $\overline{\theta}$, such that $\overline{\theta} x = \theta x$ for all $x \in Var(t)$.*

Proof. Let $\{x_1, \ldots, x_n\}$ be the variables of t'. We distinguish two cases for each variable x_i. First, assume x_i occurs in s' at position q. Then, we have that $\theta\sigma x_i \rightarrow^{*}_{\mathcal{R},\mu} \theta' x_i$, as $\theta\sigma x_i = \theta t|_q$ and $\theta' x_i = \theta' s'|_q$ and all positions above q are constructors in t and s'. Thus, we set $\overline{\theta} y = \theta y$ for all $y \in Var(Codom(\sigma))$ and obtain $\overline{\theta}\sigma t'|_{q'} \rightarrow^{*}_{\mathcal{R},\mu} \theta' t'|_{q'}$ for any position q' with $t'|_{q'} = x_i$. Note that if q is replacing in s', then so is q' in t'. Otherwise, $\theta\sigma x_i = \theta' x_i$.

Secondly, if x_i does not occur in s', then it does neither occur in $Dom(\sigma)$ nor in $Var(Codom(\sigma))$. Thus, we set $\overline{\theta} x_i = \theta' x_i$ and obtain $\overline{\theta}\sigma t'|_p = \theta' t'|_p$ for any position p with $t'|_p = x_i$.

Hence, $\theta\sigma s' \xrightarrow{\epsilon}_{\mathcal{P},\mu} \overline{\theta}\sigma t'$ and we have that $\overline{\theta}\sigma x \rightarrow^{*}_{\mathcal{R},\mu} \theta' x$ for all $x \in t'$ and thus $\overline{\theta}\sigma t' \rightarrow^{*}_{\mathcal{R},\mu} \theta' t'$. $\square$

Lemma 6 *Let $\mathcal{P} = (\Sigma, R)$ and $\mathcal{R} = (\mathcal{F} \uplus \mathcal{C}, R')$ be TRSs with a combined replacement map μ. If $\theta s \xrightarrow{\epsilon}_{\mathcal{P},\mu} \theta t \xrightarrow{>\epsilon}{}^{*}_{\mathcal{R},\mu} \theta' s' \xrightarrow{\epsilon}_{\mathcal{P},\mu} \theta' t'$, $\sigma t = s'$ for some substitution σ, $\overline{Var}_{\mu}(s) \cap Var^{\mu}(t) = \emptyset$ and all variables of t are contained only in constructor subterms (w.r.t. $\mathcal{R}$) (i.e. $t|_p \in Var \Rightarrow \forall q < p\colon t|_q \in (\Sigma \cup \mathcal{C}) \setminus \mathcal{F}$), then $\theta s \rightarrow^{*}_{\mathcal{R},\mu} \overline{\theta}\sigma s$ for some $\overline{\theta}$, such that $\overline{\theta} x = \theta' x$ for all $x \in Var(\sigma t)$.*

Proof. Let $\{x_1, \ldots, x_n\}$ be the variables of s. We distinguish two cases for each variable x_i. First, assume x_i occurs in t at position q. Then, we have that $\theta x_i \rightarrow^{*}_{\mathcal{R},\mu} \theta' \sigma x_i$, as $\theta x_i = \theta t|_q$, $\theta' \sigma x_i = \theta' s'|_q$ and all positions above q are constructors in t and s'. Thus, we set $\overline{\theta} y = \theta' y$ for all $y \in Var(Codomain(\sigma))$ and obtain $\theta s|_{q'} \rightarrow^{*}_{\mathcal{R},\mu} \overline{\theta}\sigma s|_{q'}$ for any position q' with $s|_{q'} = x_i$. Note that if q is replacing in t, then so is q' in s. Otherwise, $\theta x_i = \theta' \sigma x_i$.

Secondly, if x_i does not occur in t, then it does neither occur in $Dom(\sigma)$ nor in $Var(Codomain(\sigma))$. Thus, we set $\overline{\theta} x_i = \theta x_i$ and obtain $\theta s|_p = \overline{\theta}\sigma s|_p$ for any position p with $s|_p = x_i$. $\square$