Characterizing and Proving Operational Termination of Deterministic Conditional Term Rewriting Systems

Felix Schernhammer and Bernhard Gramlich

September 2009 – Revised Version

(the final version of this paper will be published in *The Journal of Logic and Algebraic Programming – Revised selected papers of NWPT 2008, Tarmo Uustalu and Jüri Vain, editors, 2009, to appear*)

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

September 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 (i.e. terms over the signature of the DCTRS). 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 [9] or ELAN [8]. Here we focus on the particularly interesting class of

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

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 [15] (cf. also [31, 33]).

When analyzing the termination behaviour of conditional TRSs, it turns out that the proof-theoretic notion of *operational termination* [24] 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* [31]).

Example 1. *Consider the following DCTRS $\mathcal{R}$.*

$$a \to b \Leftarrow a \to^* b$$

The induced rewrite relation is empty and thus in particular well-founded and decidable. Hence, $\mathcal{R}$ is effectively terminating (cf. [31]). On the other hand a naive algorithm recursively evaluating conditions of conditional rules in order to check applicability might loop. This is taken into account in the notion of operational termination. Indeed $\mathcal{R}$ is not operationally terminating.

For the analysis of operational termination of DCTRSs the equivalent property of quasi-decreasingness is usually used [24]. In [31], [30], based on the idea of *unravelings* of [26], 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 ([22]), that will be proved to be equivalent to operational termination of DCTRSs. Furthermore, we use a simple modification of Ohlebusch's transformation ([31]) 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* (i.e. terms over the signature of the DCTRS).

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 this 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 [18] (cf. also [5]), for proving termination of a CSRS on original terms, thus exploiting the given equivalence result. This approach has been implemented in the tool VMTL ([32])[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]

[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.

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. [6], [7] and [22]). Throughout the paper we assume that all CTRSs, CSRSs and TRSs (i.e., their induced reduction relations) are *finitely branching*.

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: $R_0 = \emptyset$, $R_{j+1} = \{l\sigma \to r\sigma \mid l \to r \Leftarrow s_1 \to^* t_1, \ldots, s_n \to^* t_n \in \mathcal{R} \wedge s_i\sigma \to^*_{R_j} t_i\sigma$ 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. [31], [15], 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 $\sigma : V \to \mathcal{T}(\Sigma', V)$ and every $i \in \{0, \ldots, n-1\}$:

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

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

A DCTRS $\mathcal{R} = (\Sigma, R)$ is *quasi-decreasing* [31] 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 $s_j\sigma \to^*_{\mathcal{R}} t_j\sigma$ for all $j \in \{1, \ldots, i\}$ implies $l\sigma \succ s_{i+1}\sigma$.

In [24] 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 [24].

The latter notions are related as follows ([31], [24]):

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 $\leadsto^{\mu}_{\mathcal{R}}$) is defined as $t \leadsto^{\mu}_{\mathcal{R}} s$ 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 $s = t[r]_p\theta$. 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 7 and Definition 13 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] , cf. also [3, 2, 23]) 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 appears non-μ-replacing in the 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]) of a CSRS $(\mathcal{R}, \mu)$, denoted $DP(\mathcal{R}, \mu)$, is $DP_o(\mathcal{R}, \mu) \cup DP_u(\mathcal{R}, \mu)$ where

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

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

- $\{l^\sharp \to D^\sharp(x) \mid l \to r \in R, x \in \mathit{Var}^\mu(r) - \mathit{Var}^\mu(l)\}$,
- $D^\sharp(f(x_1, \ldots, x_i, \ldots, x_n)) \to D^\sharp(x_i)$ for every function symbol f of any arity n and every $1 \le i \le n$ where f hides position i, and
- $D^\sharp(t) \to t^\sharp$ 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 $t_i\sigma \to^*_{\mathcal{R},\mu} s_{i+1}\sigma$ 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]).

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 4 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 $\sigma \colon V \to \mathcal{T}(\Sigma, V)$ and every $i \in \{0, \ldots, n-1\}$:*

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

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 4).

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 [31], which in turn was inspired by [26, 27].[3]

Definition 2 (unraveling of DCTRSs, [31])**.** *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:*

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

Here $\mathit{Var}(s)$ denotes an arbitrary but fixed sequence of the set of variables of the term s. Let $\mathcal{E}\mathit{Var}(t_i)$ be $\mathit{Var}(t_i) \setminus (\mathit{Var}(l) \cup \bigcup_{j=1}^{i-1} \mathit{Var}(t_j))$. By abuse of notation, by $\mathcal{E}\mathit{Var}(t_i)$ we denote an arbitrary but fixed sequence of the variables in the set $\mathcal{E}\mathit{Var}(t_i)$. 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.

[3] Note that there exist also various other transformations from conditional to unconditional TRSs in the literature, cf. e.g. [4], [34] and [19] for more recent ones. However, for our purpose of analyzing operational termination of conditional systems, the chosen transformation appears to be the most appropriate one, as other transformations typically focus on efficiency in the simulation of conditional reductions and are thus more complicated and less suitable for termination analysis.

Definition 3. *The mapping* $\mathsf{tb}\colon \mathcal{T}(U(\Sigma), V) \to \mathcal{T}(\Sigma, V)$ *(read "translate back") which is equivalent to Ohlebusch's mapping* $\bigtriangledown$ *([31, Definition* 7.2.53*]) is defined by*

$$\mathsf{tb}(t) = \begin{cases} x & \text{if } t = x \in V \\ f(\mathsf{tb}(t_1), \ldots, \mathsf{tb}(t_l)) & \text{if } t = f(t_1, \ldots, t_l) \\ & \quad \text{and } f \in \Sigma \\ l\sigma & \text{if } t = U_j^\alpha(v_1, v_2, \ldots, v_{k+1}, \ldots, v_{m_j}) \\ & \quad \text{and } \alpha = l \to r \Leftarrow c \end{cases}$$

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

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

Example 2. *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* $\mathsf{tb}(t) = f(b)$ *and clearly* $f(b) \not\to^*_{U(\mathcal{R})} U(a, b)$.

Informally, the term $t = U_j^\alpha(v_1, \ldots, v_{m_j})$ 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 suitable for verifying quasi-reductivity by proving termination of a TRS, as whenever the transformed system $U(\mathcal{R})$ is terminating, the original DCTRS $\mathcal{R}$ is *quasi-reductive* [31]. However, the converse implication does not hold.

Example 3. *([26]) Consider the DCTRS* $\mathcal{R} = (\Sigma, R)$ *given by*

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

The system $U(\mathcal{R}) = (U(\Sigma), U(R))$ is given by $U(\Sigma) = \Sigma \cup \{U_1^\alpha\}$ and $U(R) = R$ except that rule α is replaced by the rules $f(x) \to U_1^\alpha(x, x)$ and $U_1^\alpha(e, x) \to x$. $\mathcal{R}$ is quasi-reductive (and thus operationally terminating) (cf. Example 7, below), nevertheless $U(\mathcal{R})$ is non-terminating ([31]).

Roughly speaking, the problem in Example 3 is that subterms at the second position of U_1^α 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., [10], [29], [11]) with slight differences.[4]

Definition 4. *(context-sensitive unraveling of a DCTRS) Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS. The context-sensitive rewrite system $U_{cs}(\mathcal{R})$ 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 $U_{cs}(\mathcal{R})$". Moreover, we omit an explicit reference to the replacement map $\mu_{U_{cs}(\mathcal{R})}$ if it is clear from the context, for instance in $\to_{U_{cs}(\mathcal{R})}$ reductions.

Consider the following reduction:

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

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. Context-sensitivity assures that σ_i and σ_{i+1} are not contradictory, i.e., $x\sigma_i = x\sigma_{i+1}$ for all $x \in Dom(\sigma_i) \cap Dom(\sigma_{i+1})$.

Observation 1. *Let $\mathcal{R}$ be a DCTRS. For every reduction*

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

it holds that $x\sigma_i = x\sigma_{i+1}$ 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 $\mathsf{tb}(t) \to^*_{U_{cs}(\mathcal{R})} t$ provided that t is reachable by *any* term $s \in \mathcal{T}(\Sigma, V)$ (see Corollary 2, below). This is in general not true, if context-sensitivity is dropped.

[4]See Section 7 for more details.

Example 4. *Let $\mathcal{R} = (\Sigma, R)$ be the DCTRS of Example 2 extended by two unconditional rules*

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

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

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

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

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

However, it is obviously not reachable from $\mathsf{tb}(t) = f(c)$ *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 $s \in \mathcal{T}(\Sigma, V)$ (i.e. $s \to^*_{U_{cs}(\mathcal{R})} t$) coincides with reachability of t from* $\mathsf{tb}(t)$ *(cf. Corollary 2 below).*

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 $\mathsf{tb}(t)$ if t is part of reduction sequence issuing from a term of $\mathcal{T}(\Sigma, V)$, will be used extensively in the proofs of some of the main results of this paper (e.g. Theorems 1 and 4).

Certain $U_{cs}(\mathcal{R})$-reduction steps inside a U-term t will have no effect on the result of the function tb, i.e., $t \to s$ and $\mathsf{tb}(t) = \mathsf{tb}(s)$. This motivates the definition of tb-preserving reduction steps in $U_{cs}(\mathcal{R})$. 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 also yields the same result as $\mathsf{tb}(t)$. 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 $\mathsf{tb}(t) = \mathsf{tb}(s)$ if t is not a U-term, s is a U-term and $t \to s$.

Definition 5 (tb-preserving reduction steps)**.** *Let $\mathcal{R}$ be a DCTRS ($\mathcal{R} = (\Sigma, R)$) and $U_{cs}(\mathcal{R})$ its transformed CSRS. A step $s \xrightarrow{p}_{U_{cs}(\mathcal{R})} t$[5] 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.*

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

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

Proposition 3. *Let $\mathcal{R}$ be a DCTRS. If $s, t \in \mathcal{T}(U(\Sigma), V)$ and $s \to_{U_{cs}(\mathcal{R})} t$ with a tb-preserving step, then $\mathsf{tb}(s) = \mathsf{tb}(t)$.*

Proof. If the reduction step from s to t, say at position p, occurs strictly inside a U-term, then it occurs strictly inside the first argument of some maximal U-rooted subterm u at position $q < p$ in s. According to Definition 3 we have $\mathsf{tb}(s|_q) = \mathsf{tb}(t|_q)$ and thus $\mathsf{tb}(t) = \mathsf{tb}(s)$.

Otherwise, $t|_p$ is a U-term, this means that either a rule of the shape $l \to U(c, \vec{x})$ or a rule of the shape $U(c_1, \vec{x}_i) \to U_{i+1}(c_2, \vec{x}_{i+1})$ was applied. So $s|_p = l\sigma$ or $s|_p = U(c_1, \vec{x}_i)\sigma$ and $t|_p = U(c, \vec{x})\sigma$. Hence, according to Definition 3 we have $\mathsf{tb}(s|_p) = \mathsf{tb}(t|_p)$ and thus $\mathsf{tb}(s) = \mathsf{tb}(t)$. $\square$

Example 5. *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:

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

while the following one is not:

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

Before investigating the effects of using context-sensitivity in the unraveling transformation of Definition 4 on the power of proving operational termination, let us consider the capability of $U_{cs}(\mathcal{R})$ to simulate reductions of a DCTRS $\mathcal{R}$. 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 [29] it was shown that simulation soundness is obtained for their version of the transformation 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, for our transformation this additional "membership condition" is not needed (see also Section 7 below for further details).

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 $s, t \in \mathcal{T}(U(\Sigma), V)$ we have: If $s \to^*_{U_{cs}(\mathcal{R})} t$ and s is reachable from an original term (i.e., $s' \to^*_{U_{cs}(\mathcal{R})} s$ for some $s' \in \mathcal{T}(\Sigma, V)$), then $\mathsf{tb}(s) \to^*_{\mathcal{R}} \mathsf{tb}(t)$. Moreover, if $s, t \in \mathcal{T}(\Sigma, V)$, then $s \to^+_{U_{cs}(\mathcal{R})} t$ implies $s \to^+_{\mathcal{R}} t$.*

Before proving Theorem 2 we need two auxiliary lemmas. The first one (Lemma 1 below) states that whenever we have a $U_{cs}(\mathcal{R})$-reduction sequence D of the shape

$$\begin{array}{lll} s_1 & \to^*_{U_{cs}(\mathcal{R})} & s_2[l\sigma]_{p_1} \\ & \to_{U_{cs}(\mathcal{R})} & s_2[U_1^\alpha(s_1, \vec{x}_1)\sigma]_{p_1} \\ & \to^*_{U_{cs}(\mathcal{R})} & s_3[U_1^\alpha(t_1, \vec{x}_1)\sigma']_{p_2} \\ & \to_{U_{cs}(\mathcal{R})} & s_3[U_2^\alpha(s_2, \vec{x}_2)\sigma']_{p_2} \\ & \to^*_{U_{cs}(\mathcal{R})} & \cdots \\ & \to^*_{U_{cs}(\mathcal{R})} & s_{n+1}[U_n^\alpha(t_n, \vec{x}_n)\sigma^n]_{p_n} \\ & \to_{U_{cs}(\mathcal{R})} & s_{n+1}[r\sigma^n]_{p_n} \\ & \to^*_{U_{cs}(\mathcal{R})} & s_{n+2}, \end{array}$$

where s_1 is an original term which means that D contains the complete simulation of the application of a conditional rule $\alpha : l \to r \Leftarrow s_1 \to^* t_1, \ldots s_n \to^* t_n$, the reductions satisfying its conditions $s_i\sigma^n \to^*_{U_{cs}(\mathcal{R})} t_i\sigma^n$ occur as subreductions of D for all $i \in \{1, \ldots, n\}$.

The second auxiliary result (Lemma 2 below) will be crucial for the overall inductive proof structure of Theorem 2 to work.

For the former we first introduce some terminology for tracing subterms (especially U-subterms) in reduction sequences, in a forward and backward manner. In the above reduction sequence D the positions p_i mark *descendants* of the subterms $s_{i-1}|_{p_{i-1}}$ of s_{i-1}. More formally, the set of *one-step descendants* of a subterm position p of t w.r.t. a (one-step) reduction $t = C[s]_p \xrightarrow{q} t'$ is the set of subterm positions in t' given by

- $\{p\}$, if $q \geq p$ or $q \mid\mid p$,
- $\{q.o'.p' \mid t|_q = l\sigma, l|_o \in \mathit{Var}, q.o.p' = p, l|_o = r|_{o'}\}$, if $q < p$ and (a superterm of) s is bound to a variable in the matching of $t|_q$ with the left-hand side of the applied rule, and
- $\emptyset$, otherwise.

Slightly abusing terminology, when $t = C[s]_p \xrightarrow{q} t'$ with set $\{p_1, \ldots, p_k\}$ of one-step descendants in t', we also say that $t|_p$ has descendants $t'|_{p_i}$ in t'. The *descendant relation* (w.r.t given derivations) is obtained as the (reflexive-)transitive closure of the one-step descendant relation. Note that the set of one-step descendants of a U-subterm (w.r.t. a one-step derivation) is non-empty unless the subterm is erased by an erasing rule (i.e., a rule $l \to r$ such that $x \in Var(l) \setminus Var(r)$), because U-symbols occur only at but not below the root position in left-hand sides of rules of systems obtained by the transformation of Definition 4. The notions of *one-step* (and many-step) *antecedents* of a subterm position (w.r.t. a given reduction sequence) are defined analogously (in a backward manner).

Note that with a similar argument as for the existence of descendants of U-subterms we get that every U-subterm has at least one one-step antecedent w.r.t. every (one-step) reduction sequence.

Now we can express the notion of a complete simulated rule application more formally. By a complete simulated rule application we mean that all rules obtained by transforming one conditional rule are eventually applied to a certain subterm and its descendants during the reduction sequence in the right order. Yet, these (unconditional) rule applications need not be consecutive.

Note also that it makes sense to talk about descendants of U-subterms, because they can only be copied, eliminated or duplicated but not otherwise modified by more outer reductions. This is due to the special shape of the rules in systems obtained by the transformation of Definition 4. More precisely, it is due to the fact that U-symbols occur only at, but not below the root of left-hand sides of rules.

Lemma 1. *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS and let $\alpha : l \to r \Leftarrow c_1^l \to^* c_1^r, \dots c_n^l \to^* c_n^r$ be a rule from R. Moreover, assume that $D\colon s \to^*_{U_{cs}(\mathcal{R})} t$ is a non-empty $U_{cs}(\mathcal{R})$-reduction such that $s \in \mathcal{T}(\Sigma, V)$ and the last step is due to an application of the rule $U_n^\alpha(c_n^r, \vec{x}_n) \to r \in U_{cs}(\mathcal{R})$ ($r \in \mathcal{T}(\Sigma, V)$) with a substitution σ^n. Then, the reductions $C_i : c_i^l\sigma^n \to^*_{U_{cs}(\mathcal{R})} c_i^r\sigma^n$ occur as subreductions of D[6] for every $i \in \{1, \dots, n\}$.*

Proof. Let the last step of D be $t' \xrightarrow{p}_{U_{cs}(\mathcal{R})} t$. Hence, $t'|_p$ must be a U-term.

We identify the first term s' in D such that

1. s' contains at least one antecedent of $t'|_p$,
2. all antecedents of $t'|_p$ in s' are U-terms, and
3. conditions (1) and (2) also hold for all terms occurring later (but before t') in D.

Note that t' itself has the demanded properties. Thus the existence of s' is guaranteed. We now claim:

$$\text{Some antecedent } s'|_q \text{ of } t'|_p \text{ has the form } U_1^\alpha(c_1^l, \vec{x}_1)\sigma^1. \tag{1}$$

In order to show (1) assume s' did not contain a subterm of this shape. Then, consider s^0 which is the subterm preceding s' in D (this subterm exists as s' contains U-terms but s does not, so $s \neq s'$). The term s^0 contains antecedents

[6] not necessarily consecutively, and embedded in some surrounding context, i.e. they can be obtained by "extraction" from D

of $t'|_p$, because s' contains antecedents of $t'|_p$ which are U-terms. This in turn implies the existence of one-step antecedents in s^0.

Assume some antecedent of $t'|_p$ in s^0 is not a U-term. As this term has a one-step descendant in s' being in turn a antecedent of $t'|_p$ and thus a U-term, this very U-term in s' must be of the shape $U_1^\alpha(c_1^l, \vec{x}_1)\sigma^1$ as it must have been introduced by an application of the rule $l \to U_1^\alpha(c_1^l, \vec{x}_1)$. This contradicts our assumption that (1) does not hold for s'. Thus all antecedents of $t'|_p$ in s^0 must be U-terms.

This in turn contradicts the minimality of s', i.e. being the first term on D containing only U-term antecedents of $t'|_p$. Hence, we derived a contradiction from $\neg(1)$. This concludes the proof of Claim 1.

Let $s'|_q = U_1^\alpha(c_1^l, \vec{x}_1)\sigma^1$. By our choice of s' and the fact that $s'|_q$ is a antecedent of $t'|_p$, every term between s' and t' in D contains a descendant of $s'|_q$ which is also a antecedent of $t'|_p$ and a U-term.

Some descendant (of $s'|_q$) must be of the shape $U_1^\alpha(c_1^r, \vec{x}_1)\sigma'^1$, because otherwise $t'|_p$ could not be reached (cf. Definition 4). We inspect D between s' and s'' where s'' contains such a descendant of $s'|_q$ say at position q'. Then, $s'|_q$ and its descendants which are also antecedents of $s''|_{q'}$ are only (syntactically) modified by rule applications below their roots. The reason is that a term rooted by some U-symbol U_i cannot be reduced to another term having the same root symbol with reduction steps containing at least one root step, unless the reduction sequence contains a non-U-term (cf. Definition 4).

Hence, we can extract the reduction $c_1^l\sigma^1 \to^*_{U_{cs}(\mathcal{R})} c_1^r\sigma'^1$ from D.

The same argumentation applies also to all other conditions as $U_i^\alpha(c_i^l, \vec{x}_i)\sigma^i$ must occur (by our choice of s' and q), as descendants of $s'|_q$ and antecedents of $t'|_p$ in D (in particular in such a way that σ^i does not contradict σ'^{i-1}). Moreover, by Observation 1 the used substitutions are not contradictory and their domains are subdomains of the one of σ^n, which is due to the fact that the set of variables stored by a U-symbol U_i^α is a subset of the ones stored by U_j^α provided that $i \leq j$ (cf. Definition 4). □

The second lemma states that if there exists a parallel reduction sequence $s \mathrel{+\!\!\!\!\!\!\;\mapsto}^*_{U_{cs}(\mathcal{R})} t$, where s is an original term, then for all positions p of t there is also a parallel reduction $s' \mathrel{+\!\!\!\!\!\!\;\mapsto}^*_{U_{cs}(\mathcal{R})} t|_p$ for some original term s' such that its length is less than or equal to the length of the former parallel reduction sequence.

In order to formalize this proposition we introduce the notion of the *minimal parallel* Σ*-distance* of a term (over $\mathcal{T}(U(\Sigma), V)$) (from any original term).

Definition 6 (minimal parallel Σ-distance)**.** *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS and $t \in \mathcal{T}(U(\Sigma), V)$. The* minimal parallel Σ-distance *of t (w.r.t. a DCTRS $\mathcal{R}$) is given by*

$$mpd_\Sigma(t) = inf\{n \mid \exists s \in \mathcal{T}(\Sigma, V).s \mathrel{+\!\!\!\!\!\!\;\mapsto}^n_{U_{cs}(\mathcal{R})} t\}$$

where $\mathrel{+\!\!\!\!\!\!\;\mapsto}^n_{U_{cs}(\mathcal{R})}$ means that n parallel reductions are performed.

Note that $inf\ \emptyset = +\infty$, so the minimal parallel Σ-distance of any term t that is not reachable from an original term is $+\infty$. Note on the other hand that if t is reachable from an original term, then the inf in Definition 6 is actually a min, as lengths of reductions are natural numbers and hence we can find a concrete (parallel) reduction from an original term to t with length $mpd_\Sigma(t)$.

Lemma 2. *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS and $t \in \mathcal{T}(U(\Sigma), V)$ with $mpd_\Sigma(t) < \infty$. Then for every subterm $t|_p$ of t we have $mpd_\Sigma(t|_p) \leq mpd_\Sigma(t)$. Moreover, if $t|_p$ occurs strictly inside a U-term in t, then $mpd_\Sigma(t|_p) < mpd_\Sigma(t)$.*

Proof. Let $D\colon u \mathrel{+\!\!\!\!\!\!\mapsto}^*_{U_{cs}(\mathcal{R})} t$ where $u \in \mathcal{T}(\Sigma, V)$ be a reduction sequence of length $mpd_\Sigma(t)$. We prove the result by induction on $mpd_\Sigma(t)$. If $mpd_\Sigma(t) = 0$, the result holds vacuously as t is an original term and thus every subterm of t is an original term as well.

Assume $mpd_\Sigma(t) = m$, then we can write D as

$$u \mathrel{+\!\!\!\!\!\!\mapsto}^{m-1}_{U_{cs}(\mathcal{R})} t' \mathrel{+\!\!\!\!\!\!\mapsto}_{U_{cs}(\mathcal{R})} t.$$

We consider the maximal U-rooted subterms $u_1, \ldots, u_n$ of $t|_p$ s.t.

$$t|_p = C[u_1, \ldots u_n]_{p_1,\ldots,p_n}.$$

Each subterm u_i has at least one one-step antecedent u_i' in t' and the induction hypothesis yields that $mpd_\Sigma(u_i') \leq mpd_\Sigma(t') \leq m-1$. Hence, as we are using parallel reduction we obtain

$$mpd_\Sigma(C[u_1', \ldots u_n']_{p_1,\ldots,p_n}) \leq m-1. \tag{2}$$

Moreover, we have $u_i' \mathrel{+\!\!\!\!\!\!\mapsto}^*_{U_{cs}(\mathcal{R})} u_i$ with zero or one reduction steps and as all u_i are parallel in $t|_p$ we have $C[u_1', \ldots u_n']_{p_1,\ldots,p_n} \mathrel{+\!\!\!\!\!\!\mapsto}^*_{U_{cs}(\mathcal{R})} C[u_1, \ldots u_n]_{p_1,\ldots,p_n} = t|_p$ with zero or one steps. Thus, $mpd_\Sigma(t|_p) \leq m$.

Now assume that $t|_p$ occurs strictly inside a U-term in t. We distinguish two cases.

- If $u_i' \neq u_i$ for some $i \in \{1, \ldots, n\}$ (i.e. if some reduction step from t' to t occurred inside a antecedent of some u_i), then by the definition of the descendant relation and the shape of the rules in $U_{cs}(\mathcal{R})$, i.e. the fact that U-symbols occur only at but not below the root of left- and right-hand sides of all rules, we get that if u_i occurs at position $q \geq p$ in t, then u_i' occurs at position q in t' and there must have been a reduction in $t' \mathrel{+\!\!\!\!\!\!\mapsto}_{U_{cs}(\mathcal{R})} t$ at or below q. Hence, in the same parallel step there was no reduction above p and thus all u_i' occur inside a U-term in t' as they occur inside $t'|_p$.

  Hence, the induction hypothesis yields $mpd_\Sigma(u_i') \leq m-2$ and we get $mpd_\Sigma(t|_p) \leq m-1$.

- Otherwise, if $u_i' = u_i$ for all $i \in \{1, \ldots, n\}$, (2) and $C[u_1', \ldots u_n']_{p_1,\ldots,p_n} = t|_p$ yield $mpd_\Sigma(t|_p) \leq m-1$.

$\square$

Indeed Lemma 2 does not hold if one considers ordinary $U_{cs}(\mathcal{R})$-reduction instead of parallel reduction.

Example 6. *Consider the following one-rule DCTRS $\mathcal{R}$*

$$f(x) \to b \Leftarrow g(x,x) \to^* a$$

$U_{cs}(\mathcal{R})$ is given by

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

Now consider the following $U_{cs}(\mathcal{R})$ reduction sequence of length 2

$$\begin{array}{rcl} f(f(x)) & \to & f(U(g(x,x),x)) \\ & \to & U(g(U(g(x,x),x),U(g(x,x),x)),U(g(x,x),x)) = t \end{array}$$

However, it is easy to see that at least 2 reduction steps are necessary to derive

$$t|_1 = g(U(g(x,x),x),U(g(x,x),x))$$

from an original term although it occurs as subterm strictly below a U-symbol in t.

Now we are ready to prove Theorem 2.

Proof of Theorem 2. For the first part of the theorem, we prove the equivalent result that $s \mathrel{+\!\!\!\!\!\!\;+\!\!\!\!\rightarrow}^*_{U_{cs}(\mathcal{R})} t$ implies $\mathsf{tb}(s) \mathrel{+\!\!\!\!\!\!\;+\!\!\!\!\rightarrow}^*_{\mathcal{R}} \mathsf{tb}(t)$ provided that $s,t \in \mathcal{T}(U(\Sigma),R)$ and s is reachable from an original term.

In order to prove this by induction, we associate to each reduction sequence $S\colon s \mathrel{+\!\!\!\!\!\!\;+\!\!\!\!\rightarrow}^*_{U_{cs}(\mathcal{R})} t$ with $s,t \in \mathcal{T}(U(\Sigma),V)$ a non-negative integer (its order) k where $k = mpd_\Sigma(s) + l$ and l is the length (i.e. the number of parallel reduction steps) of S. We use induction over k (note that $mpd_\Sigma(s)$ and l are both non-negative for every reduction sequence S).

For the base case (i.e., $k=0$) the theorem holds trivially, since $s=t$. For the inductive step, consider a reduction sequence $S : s \mathrel{+\!\!\!\!\!\!\;+\!\!\!\!\rightarrow}^*_{U_{cs}(\mathcal{R})} s' \mathrel{+\!\!\!\!\!\!\;+\!\!\!\!\rightarrow}_{U_{cs}(\mathcal{R})} t$. The induction hypothesis yields $\mathsf{tb}(s) \to^*_{\mathcal{R}} \mathsf{tb}(s')$. Thus, for $\mathsf{tb}(s) \to^*_{\mathcal{R}} \mathsf{tb}(t)$ it suffices to show that

$$\mathsf{tb}(s') \to^*_{\mathcal{R}} \mathsf{tb}(t) \tag{3}$$

holds.

We prove this by (nested) induction over the number of single (non-parallel) reduction steps in $s' \mathrel{+\!\!\!\!\!\!\;+\!\!\!\!\rightarrow}_{U_{cs}(\mathcal{R})} t$. If this number is zero, then $s' = t$ and thus $\mathsf{tb}(s') = \mathsf{tb}(t)$.

Otherwise, we split $s' \mathrel{+\!\!\!\!\!\!\;+\!\!\!\!\rightarrow}_{U_{cs}(\mathcal{R})} t$ into $s' \mathrel{+\!\!\!\!\!\!\;+\!\!\!\!\rightarrow}_{U_{cs}(\mathcal{R})} t' \to_{U_{cs}(\mathcal{R})} t$ and the induction hypothesis yields $\mathsf{tb}(s') \to^*_{\mathcal{R}} \mathsf{tb}(t')$.

We distinguish 3 cases depending on the reduction from t' to t.

1. Assume the step is tb-preserving. Then we have $\mathsf{tb}(t') = \mathsf{tb}(t)$ and hence $\mathsf{tb}(s') \to_{\mathcal{R}}^* \mathsf{tb}(t)$, i.e., (3).

2. If the step is non-tb-preserving and using a rule $l \to r$ which already occurred in the DCTRS (i.e. as unconditional rule) say at position p, then $t'|_p = l\sigma$. As the reduction is non-tb-preserving, there is no U-symbol in t' above p (cf. Definition 5). Moreover, there are no U-symbols in l (as it already occurred in $\mathcal{R}$), hence $\mathsf{tb}(t')|_{p.q} = \mathsf{tb}(t'|_{p.q})$ for all variable positions q of l, i.e. $\mathsf{tb}(t')|_p = l\sigma'$ and $x\sigma' = \mathsf{tb}(x\sigma)$ for all $x \in Dom(\sigma)$. Thus, $\mathsf{tb}(t') = \mathsf{tb}(t')[l\sigma']_p \to_{\mathcal{R}} \mathsf{tb}(t')[r\sigma']_p = \mathsf{tb}(t)$, and finally (3).

3. Assume the step (at position p) is non-tb-preserving and using a rule $U(u, x_1, \ldots, x_o) \to r$ where $root(r) \in \Sigma$ (say $t'|_p = U(u, x_1, \ldots, x_o)\sigma)$. This rule stems from a conditional rule $\alpha : l \to r \Leftarrow c_1^l \to^* c_1^r, \ldots c_m^l \to^* c_m^r \in \mathcal{R}$.

   In order to perform the corresponding reduction in the conditional system $\mathcal{R}$, we need to make sure that $\mathsf{tb}(c_i^l\sigma) \to_{\mathcal{R}}^* \mathsf{tb}(c_i^r\sigma)$ holds for every $i \in \{1, \ldots, m\}$.

   We consider the following reduction sequence S' in $U_{cs}(\mathcal{R})$

   $$S' \colon u \mathrel{+\!\!\!\!\!\;\!\!\vdash\!\!\!\!\rightarrow}^*_{U_{cs}(\mathcal{R})} s \mathrel{+\!\!\!\!\!\;\!\!\vdash\!\!\!\!\rightarrow}^*_{U_{cs}(\mathcal{R})} s'$$

   where u is some *original* term such that the length of the reduction from u to s is exactly $mpd_\Sigma(s)$. Note that $s'|_p = U(u, x_1, \ldots, x_o)\sigma)$ because all reduction steps from s' to t' were parallel to p.

   The existence of S' ensures that for each condition $c_i^l \to^* c_i^r$ the reduction $c_i^l\sigma \mathrel{+\!\!\!\!\!\;\!\!\vdash\!\!\!\!\rightarrow}^*_{U_{cs}(\mathcal{R})} c_i^r\sigma$ occurred as subreduction of S', by Lemma 1.

   Consider a term $c_i^l\sigma$ occurring as a subterm of some term v in S'. We partition the reduction sequence S' in reduction steps that happen before v (which we call the head of S') and in reduction steps happening after v (which we call the tail of S').

   The reduction from $c_i^l\sigma$ to $c_i^r\sigma$ is part of the tail of S' and thus its (parallel) length is not longer than this tail. Moreover, Lemma 2 yields that $mpd_\Sigma(c_i^l\sigma)$ is shorter than the head of S', because $c_i^l\sigma$ occurs inside a U-term. Hence, the order of the reduction sequence $c_i^l\sigma \mathrel{+\!\!\!\!\!\;\!\!\vdash\!\!\!\!\rightarrow}^*_{U_{cs}(\mathcal{R})} c_i^r\sigma$ is smaller than (or equal to) the length of the reduction sequence S' which is exactly the order of the reduction from s to s' and thus smaller than the order of our initial reduction sequence S. Hence, the induction hypothesis (of the outer induction) applies yielding $\mathsf{tb}(c_i^l\sigma) \to_{\mathcal{R}}^* \mathsf{tb}(c_i^r\sigma)$ for all $i \in \{1, \ldots, m\}$.

   Now consider $t' = t'[U(u, x_1, \ldots, x_o)\sigma]_p$. Let $\tau = \mathsf{tb}(\sigma)$, i.e. $x\tau = \mathsf{tb}(x\sigma)$ for all $x \in Dom(\sigma)$. Then we have $\mathsf{tb}(t') = \mathsf{tb}(t')[l\tau]_p$. And since $c_i^l\tau \to_{\mathcal{R}}^* c_i^r\tau$ for all $i \in \{1, \ldots, m\}$, we finally obtain $\mathsf{tb}(t')[l\tau]_p \to_{\mathcal{R}} \mathsf{tb}(t')[r\tau]_p = \mathsf{tb}(t)$.

This concludes the inner induction and also the outer step case.

Note that, in the inner induction above, if not all steps in a reduction sequence S are tb-preserving, i.e. whenever items (2) or (3) apply, then the corresponding sequence in the conditional system is non-empty. Hence, whenever $s \to^{+}_{U_{cs}(\mathcal{R})} t$ and $s, t \in \mathcal{T}(\Sigma, V)$, then $\mathsf{tb}(s) \to^{+}_{\mathcal{R}} \mathsf{tb}(t)$ is non-empty, too. $\square$

Next we show that for any term t that is reachable from an original one, say s, the corresponding reduction can be factored through $\mathsf{tb}(t)$ such that the first part only uses $\mathcal{R}$-steps and the latter one only tb-preserving $U_{cs}(\mathcal{R})$-steps.

Lemma 3. *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS. If a term $t \in \mathcal{T}(U(\Sigma), V)$ is reachable from an original term (i.e., if $mpd_\Sigma(t) < \infty$), then we have $\mathsf{tb}(t) \to^{*}_{U_{cs}(\mathcal{R})} t$ with tb-preserving steps.*

Proof. We prove the result by induction on $mpd_\Sigma(t)$. If $mpd_\Sigma(t) = 0$, then t is an original term and the result is immediate.

Otherwise, let $mpd_\Sigma(t) = n > 0$. Then, there is a parallel $U_{cs}(\mathcal{R})$-reduction sequence $D\colon u \mathrel{+\!\!\!\!\!\!\longrightarrow}^{n-1}_{U_{cs}(\mathcal{R})} t' \mathrel{+\!\!\!\!\!\!\longrightarrow}_{U_{cs}(\mathcal{R})} t$ of length n. Let $u_1, \ldots, u_m$ be the maximal U-rooted subterms of t s.t.

$$t = C[u_1, \ldots, u_m]_{p_1,\ldots,p_m}$$

Each u_i has one or several one-step antecedents u_i^j (in t') for $j \in \{1, \ldots, k_i\}$, where k_i is the number of one-step antecedents of u_i in D. For all $i \in \{1, \ldots, m\}$ and all $j \in \{1, \ldots, k_i\}$ $mpd_\Sigma(u_i^j) < n$ by Lemma 2, hence the induction hypothesis yields $\mathsf{tb}(u_i^j) \to^{*}_{U_{cs}(\mathcal{R})} u_i^j$ with tb-preserving steps.

Moreover, we get $u_i^j \to_{U_{cs}(\mathcal{R})} u_i$ for all $i \in \{1, \ldots, m\}$ and all $j \in \{1, \ldots, k_i\}$, and these steps are tb-preserving, because the u_i's are U-terms. Hence we obtain $\mathsf{tb}(u_i) = \mathsf{tb}(u_i^j) \to^{*}_{U_{cs}(\mathcal{R})} u_i^j \to_{U_{cs}(\mathcal{R})} u_i$ with tb-preserving steps and as the u_i's are the maximal U-rooted terms in t, we finally get

$$\mathsf{tb}(t) = C[\mathsf{tb}(u_1), \ldots, \mathsf{tb}(u_m)]_{p_1,\ldots,p_m} \to^{*}_{U_{cs}(\mathcal{R})} C[u_1, \ldots, u_m]_{p_1,\ldots,p_m} = t$$

with only tb-preserving steps. $\square$

Corollary 2. *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS. Whenever $s \to^{*}_{U_{cs}(\mathcal{R})} t$ and $t, s \in \mathcal{T}(U(\Sigma), V)$ where s is reachable from an original term, then $\mathsf{tb}(s) \to^{*}_{U_{cs}(\mathcal{R})} \mathsf{tb}(t) \to^{*}_{U_{cs}(\mathcal{R})} t$, such that $\mathsf{tb}(t) \to^{*}_{U_{cs}(\mathcal{R})} t$ consists only of tb-preserving steps.*

Proof. Immediate consequence of Theorems 2, 1 and Lemma 3. $\square$

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 $s_j\sigma \succeq_\mu t_j\sigma$ 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 $U_{cs}(\mathcal{R})$:

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

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

The following corollary (of Theorem 3 and Corollary 1) has already been proved in [11].

Corollary 3. *([11]) 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 4. *([11]) Let $\mathcal{R}$ be a DCTRS. If $U(\mathcal{R})$ is terminating, then $U_{cs}(\mathcal{R})$ is μ-terminating.*

Example 7. *Consider the DCTRS $\mathcal{R}$ of Example 3. The transformed system $U_{cs}(\mathcal{R})$ (which is identical to the non-terminating TRS $U(\mathcal{R})$, 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 12 below).*

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

Example 8. *([31, 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., [31]). 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).

Definition 7 (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 sequence issuing from a term $t \in \mathcal{T}(\Sigma, V)$ in $\mathcal{R}$.*

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 $D\colon s \to \ldots$ 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 $s \to^* t \xrightarrow{\epsilon} u \to \ldots$ 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, $t = l\sigma$, $u = r\sigma$. According to Corollary 2 we have $s \to^* \mathsf{tb}(t) \to^* t$. Since $t = l\sigma$, we get $\mathsf{tb}(t) = l\sigma'$, because the steps from $\mathsf{tb}(t)$ to t are tb-preserving and $x\sigma' \to^* x\sigma$ for all $x \in Dom(\sigma)$. Thus, we have $s \to^* \mathsf{tb}(t) = l\sigma' \to r\sigma' \to^* r\sigma = u$. Furthermore, by quasi-decreasingness we get $s \succ r\sigma'$ because of $\to_{\mathcal{R}} \subseteq \succ$ and $s \to^+ r\sigma' \Rightarrow s \to^+_{\mathcal{R}} r\sigma' \in \mathcal{T}(\Sigma, V)$ (according to Theorem 2). This means that in every infinite reduction sequence starting from s we eventually arrive at $r\sigma' \prec s$, which hence also initiates an infinite $\rightarrow$-reduction, thus yielding a smaller counterexample (since $s \succ r\sigma'$). But this contradicts our minimality assumption.

Secondly, assume the transformed version of a conditional rule $l \rightarrow r \Leftarrow s_1 \rightarrow^* t_1, \ldots, s_n \rightarrow^* t_n$ is applied to t. Hence, $t = l\sigma$ and as before we get $\mathsf{tb}(t) = l\sigma'$ where $x\sigma' \rightarrow^* x\sigma$ for all $x \in Dom(\sigma)$. Thus $u = U_1(s_1, x_1, \ldots, x_{k_1})\sigma$ and we have $\mathsf{tb}(t) \rightarrow U_1(s_1, x_1, \ldots, x_{k_1})\sigma'$. By quasi-decreasingness we get $l\sigma' \succ s_1\sigma', x_1\sigma', \ldots, x_{k_1}\sigma'$, hence all the latter terms are terminating by minimality of the counterexample. Therefore, $s_1\sigma$ and $x_1\sigma, \ldots, x_{k_1}\sigma$ are terminating, too, because of $y\sigma' \rightarrow^* y\sigma$ for all $y \in Dom(\sigma)$. Thus, the only possibility of an infinite reduction from u is via a next root reduction step: $u = U_1(s_1, x_1, \ldots, x_{k_1})\sigma \rightarrow^* U_1(t_1, x_1, \ldots, x_{k_1})\sigma_1 \xrightarrow{\epsilon} U_2(s_2, x_1, \ldots, x_{k_2})\sigma_1$. So $s_1\sigma' \rightarrow^* s_1\sigma \rightarrow^* t_1\sigma_1$, and Corollary 2 yields $s_1\sigma' \rightarrow^* \mathsf{tb}(t_1\sigma_1) = t_1\sigma'_1 \rightarrow^* t_1\sigma_1$. Then it also holds that $U_1(t_1, x_1, \ldots, x_{k_1})\sigma'_1 \rightarrow U_2(s_2, x_1, \ldots, x_{k_2})\sigma'_1$ and as $s_1\sigma'_1 \rightarrow^* t_1\sigma'_1$, we have $s_1\sigma'_1 \rightarrow^*_{\mathcal{R}} t_1\sigma'_1 \in \mathcal{T}(\Sigma, V)$ according to Theorem 2 and thus $l\sigma'_1 \succ s_2\sigma'_1$. By minimality, $s_2\sigma'_1$ and $x_1\sigma'_1, \ldots, x_{k_2}\sigma'_1$ are terminating, hence also $s_2\sigma_1$ and $x_1\sigma_1, \ldots, x_{k_2}\sigma_1$ because of $x\sigma' \rightarrow^* x\sigma$ for all $x \in Dom(\sigma)$. Similarly, an infinite reduction from $U_2(s_2, x_1, \ldots, x_{k_2})\sigma_1$ is only possible via a next reduction step for which we need $s_2\sigma_1 \rightarrow^* t_2\sigma_2$ for some σ_2. By continuing this argumentation, we finally get that $l\sigma$ must eventually be reduced to $U_n(t_n, x_1, \ldots, x_{k_n})\sigma_n$ and $l\sigma'$ can be reduced to $U(t_n, x_1, \ldots, x_{k_n})\sigma'_n$. We have that $t_n\sigma'_n \in \mathcal{T}(\Sigma, V)$ is terminating by minimality (and quasi-decreasingness) and $t_n\sigma_n$ is terminating because of $t_n\sigma'_n \rightarrow^* t_n\sigma_n$. Therefore, the term $U(t_n, x_1, \ldots, x_{k_n})\sigma_n$ is reduced to $r\sigma_n$ and $U(t_n, x_1, \ldots, x_{k_n})\sigma'_n$ can be reduced to $r\sigma'_n$. We have $l\sigma'(= l\sigma'_n) \succ r\sigma'_n$ because of $l\sigma' \rightarrow^+ r\sigma'_n \in \mathcal{T}(\Sigma, V)$ and thus $l\sigma' \rightarrow^+_{\mathcal{R}} r\sigma'_n$ by Theorem 2. Hence, $r\sigma'_n$ (with $s \rightarrow^* r\sigma'_n \rightarrow^* r\sigma_n$) is terminating because of minimality and $r\sigma_n$ is terminating due to $r\sigma'_n \rightarrow^* r\sigma_n$. 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 $s \rightarrow^+_{U_{cs}(\mathcal{R})} t$ and s is reachable (in $\rightarrow_{U_{cs}(\mathcal{R})}$) by a term of the original signature (i.e. $\mathsf{tb}(s) \rightarrow^*_{U_{cs}(\mathcal{R})} s$). This relation is well-founded because $\rightarrow_{U_{cs}(\mathcal{R})}$ 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 $\succ_\mu \subseteq \rightarrow^+_{U_{cs}(\mathcal{R})}$.

First, we will deal with well-foundedness of $\succ_\mu$. Consider decreasing $\succ_\mu$-chains starting from a term t. If $s \rightarrow^*_{U_{cs}(\mathcal{R})} t$ 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 $\rightarrow_{U_{cs}(\mathcal{R})}$ on

$\mathcal{T}(\Sigma, V)$. Otherwise, $t = C[t_1 \dots t_n]_{p_1 \dots p_n}$, 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 $i \in \{1, \dots, n\}$ and the same is true for no proper superterm of any t_i. Thus, if $t \succ_\mu u$, then $u = C[t_1 \dots u_i \dots t_n]_{p_1 \dots p_i \dots p_n}$ and $t_i \succ u_i$. Furthermore, if $u \succ_\mu v$, then $v = C[t_1 \dots u_i \dots v_j \dots t_n]_{p_1 \dots p_i \dots p_j \dots p_n}$ 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 $s_i\sigma \succ_\mu t_i\sigma$ for all $1 \leq i < j$, then we get (cf., the proof of Theorem 3) $l\sigma \to^*_{U_{cs}(\mathcal{R})} U(s_j, x_1, \dots, x_m)\sigma$ and thus $l\sigma \succ^{st}_\mu s_i\sigma$ for all rules $l \to r \Leftarrow s_1 \to^* t_1, \dots, s_n \to^* t_n$, all $0 \leq j \leq n$ and all substitutions $\sigma\colon V \to \mathcal{T}(\Sigma, V)$. Analogously, if $s_i\sigma \succeq_\mu t_i\sigma$ for all $1 \leq i \leq n$, then we have $l\sigma \to^*_{U_{cs}(\mathcal{R})} r\sigma$ and thus $l\sigma \succ_\mu r\sigma$.

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

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

Corollary 4. *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 [11]. 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 8 (C_E-termination, [20, 31])**.** *We call a CSRS $\mathcal{R}$ with replacement map μ C_E-μ-*terminating *(or just C_E-terminating) if $\mathcal{R} \uplus C_E$*[7] *with $\mu(G) = \{1, 2\}$ is μ-terminating. Moreover, we define $C_E = \{G(x, y) \to x, G(x, y) \to y\}$.*

Definition 9 (C_E-cs-quasi-reductivity)**.** *Let $\mathcal{R}$ be a DCTRS. We call $\mathcal{R}$ C_E-*cs-quasi-reductive *if $\mathcal{R} \uplus C_E$ is cs-quasi-reductive.*

Lemma 4. *Let $U_{cs}(\mathcal{R})$ 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*

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

an infinite reduction sequence starting from a term t, such that $root(t) \in \Sigma$ and every replacing subterm of t is μ-terminating.

Proof. In the following we call non-μ-terminating terms containing only μ-terminating proper subterms *minimal non-terminating.*

The basic idea of the proof is to show that a minimal non-terminating term u rooted by a U-symbol must either be reduced to a minimal non-terminating term that is not rooted by a U-symbol, or it must contain a (forbidden) U-rooted minimal non-terminating proper subterm. In both cases we will derive a contradiction to the assumption that every minimal non-terminating term is rooted by a U-symbol.

Let $U_1^\alpha, \ldots, U_n^\alpha$ be the U-symbols introduced when transforming a conditional rule α (cf. Definition 2). Assume towards a contradiction that

$$U_{cs}(\mathcal{R}) \text{ is not } \mu\text{-terminating and no term } t \text{ as in the lemma exists.} \quad (4)$$

Thus, there exists a non-terminating U-term u where every replacing proper subterm of u is μ-terminating, because the existence of a non-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.

Assume $u = U_j^\alpha(u_1, \ldots, u_m)$. We first prove the following claim by induction on $n - j$ where n is the number of conditions of α.

> If u is minimal non-terminating, then the forbidden subterm u_i contains an allowed minimal non-terminating subterm for some $2 \leq i \leq m$.

First assume $u = U_n^\alpha(u_1, u_2, \ldots, u_m)$ and u is minimal non-terminating. Hence, eventually in every infinite reduction there will be a (first) root reduction step $u \to^*_{U_{cs}(\mathcal{R})} u' \xrightarrow{\epsilon}_{U_{cs}(\mathcal{R})} r\sigma$ where $r \in \mathcal{T}(\Sigma, V)$ (cf. Definition 4). From our assumption (4) it follows that $r\sigma$ must contain a minimal non-terminating U-subterm inside the substitution. The arguments $u_2, \ldots, u_m$ are forbidden for reduction in u, so for every $x \in Var(r)$ either $x\sigma$ occurred as forbidden subterm in u or it occurred allowed in u' in which case it cannot be non-terminating as u' is minimal non-terminating (obviously a minimal non-terminating term cannot be reduced to a term containing a non-terminating proper subterm by reduction steps below the root). Hence, the claim holds.

Second, assume $u = U_j^\alpha(u_1, u_2, \ldots, u_k)$ with $j < n$ and u is minimal non-terminating. In every infinite reduction sequence issuing from u there will be a (first) root reduction step

$$u \to^*_{U_{cs}(\mathcal{R})} u' \xrightarrow{\epsilon}_{U_{cs}(\mathcal{R})} u'' = U_{j+1}^\alpha(\_, u_2, \ldots, u_k, u_{k+1}, \ldots, u_{k+l}).$$

The term u'' is non-terminating (as it is part of an infinite reduction) and thus contains an allowed minimal non-terminating subterm. We distinguish two cases

- If u'' itself is minimal non-terminating, then we apply the induction hypothesis yielding that an allowed subterm of u_i is minimal non-terminating for some $i \in \{2, \ldots k+l\}$. The terms $\{u_{k+1}, \ldots, u_{k+l}\}$ occurred at allowed positions in u' (these terms are variable bindings of variables occurring in the right-hand side of the j^{th} condition of α). Thus they cannot contain a minimal non-terminating allowed subterm as this would contradict minimal non-termination of u' and thus of u. Hence, one of the terms $u_2, \ldots u_k$ contains an allowed minimal non-terminating subterm.

- If a proper subterm of u'' is minimal non-terminating, then this subterm must be in the substitution part of $r\sigma = U^{\alpha}_{j+1}(s, x_2, \ldots x_{k+l})\sigma = u''$, where r is the right-hand side of the rule applied in the root reduction, because all proper subterms of r are either variables or rooted by symbols from Σ and thus cannot be minimal non-terminating because of assumption (4). However, for every $x \in \mathit{Var}(r)$, the term $x\sigma$ already occurred in u' and as u' is minimal non-terminating, the terms $x_{k+1}\sigma, \ldots, x_{k+l}\sigma$ are terminating. Hence, an allowed subterm of $x_i\sigma$ is minimal non-terminating for some $i \in \{2, \ldots, k\}$.

Now we have shown that under assumption (4) it holds that every minimal non-terminating term contains a forbidden (and thus proper) subterm with the same property which is obviously a contradiction. Hence, assumption (4) cannot hold and the lemma is proved. □

The following definition will be useful in proving the subsequent completeness result concerning termination.

Definition 10 (partial evaluation). *Let $U_{cs}(\mathcal{R})$ be a CSRS obtained from a DCTRS $\mathcal{R} = (\Sigma, R)$ 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*

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

where $G'(g_1, \ldots, g_k)$ stands for $G(g_1, G(g_2, \ldots G(g_{k-1}, G(g_k, A)) \ldots))$ or A and the terms u_i are all terms satisfying $t \to^{+}_{U_{cs}(\mathcal{R})} u_i$ and $root(u_i) \in \Sigma \cup \mathit{Var}$, in an arbitrary but fixed order. If there is no such term then $peval(t) = A$. Here, A is a fresh constant and G is a fresh binary symbol (which will be used as non-deterministic projection symbol, i.e., by including the rules $G(x, y) \to x$, $G(x, y) \to y$, in Theorem 6 below).

Note that whenever a U-term t is μ-terminating and $t \to^{*}_{U_{cs}(\mathcal{R})} t'$, then the maximal U-rooted subterms of t' are μ-terminating as well, because they occur at replacing positions in t' since all arguments of all non-U function symbols are replacing according to Definition 4. Hence, $peval$ is well-defined.

Informally, $peval(t)$ represents all descendants of t (w.r.t. $\to_{U_{cs}(\mathcal{R})}$) that do not contain any U-symbols.

Definition 11 (correspondence w.r.t to *peval*)**.** *Let $\mathcal{R}$ be a DCTRS and $U_{cs}(\mathcal{R})$ be the system obtained by the transformation of Definition 4. Furthermore, let $s, t \in \mathcal{T}(U(\Sigma) \uplus \{G, A\}, V)$. We say that s weakly corresponds to t w.r.t. peval, denoted by $t \curvearrowright s$, if $s = C[s_1, \ldots, s_n]_{p_1,\ldots,p_n}$, $t = C[t_1, \ldots, t_n]_{p_1,\ldots,p_n}$, and for all $1 \leq i \leq n$ we have that t_i is a μ-terminating U-term with $s_i = peval(t_i)$.*

Note that the context C in Definition 11 may contain U-symbols and is unique for all terms s and t with $s \curvearrowright t$.

Lemma 5. *Let $\mathcal{R}$ be a DCTRS and let $U_{cs}(\mathcal{R})$ be the system obtained by the transformation of Definition 4. Given two terms $s, t \in \mathcal{T}(U(\Sigma) \cup \{G, A\}, V)$ with $t \curvearrowright s$, i.e. $t = C[t_1, \ldots, t_n]_{p_1,\ldots,p_n}$ and $s = C[peval(t_1), \ldots, peval(t_n)]_{p_1,\ldots,p_n}$*

1. *$t \xrightarrow{q}_{U_{cs}(\mathcal{R}) \cup C_E} t'$ and $q \geq p_i$ for some $1 \leq i \leq n$ implies $s \to^*_{U_{cs}(\mathcal{R}) \cup C_E} s'$ and $t' \curvearrowright s'$, and*
2. *$t \xrightarrow{q}_{U_{cs}(\mathcal{R}) \cup C_E} t'$ and $q < p_i$ for some $1 \leq i \leq n$ implies $s \to^+_{U_{cs}(\mathcal{R}) \cup C_E} s'$ and $t' \curvearrowright s'$.*

Proof. (1.) Let $q \geq p_j$. The term $peval(t'|_{p_j}) = G'(peval(u_1), \ldots,\ peval(u_n))$ where the set $\{u_1, \ldots, u_n\}$ is the set of all terms u_i satisfying that $t'|_{p_j} \to^*_{U_{cs}(\mathcal{R})} u_i$ and $root(u_i) \in \Sigma \cup \mathit{Var}$ according to Definition 10.

On the other hand $peval(t|_{p_j}) = G'(peval(v_1), \ldots, peval(v_m))$. Whenever $t'|_{p_j} \to^*_{U_{cs}(\mathcal{R})} u_i$, then also $t|_{p_j} \to^*_{U_{cs}(\mathcal{R})} u_i$, because $t|_{p_j} \to_{U_{cs}(\mathcal{R})} t'|_{p_j} \to^*_{U_{cs}(\mathcal{R})} u_i$. Hence, $\{u_1, \ldots, u_n\} \subseteq \{v_1, \ldots, v_m\}$ and $peval(t|_{p_j}) \to^*_{U_{cs}(\mathcal{R}) \cup C_E} peval(t'|_{p_j})$ by applying G-rules to filter those v_is that do not occur in $\{u_1, \ldots, u_n\}$. Hence, $s = s[peval(t|_{p_j})]_{p_j} \to^*_{U_{cs}(\mathcal{R}) \cup C_E} s[peval(t'|_{p_j})]_{p_j} = s'$.

(2.) Let $q < p_i$ for some $1 \leq i \leq n$. We have $t = t[l\sigma]_q$ and thus $s = s[l\sigma']_q$ because $q < p_i$ for some i, and hence $q.q' < p_i$ for all $q' \in Pos_{U(\Sigma) \cup \{G\}}(l)$ because l does not contain a U-symbol below the root and $t|_{p_i}$ is a U-term for all i. Moreover, for all $1 \leq i, j \leq n$ we have that $t_i = t_j$ implies $s_i = s_j$. Hence, l matches $s|_q$ even if it is non-linear. Obviously, $x\sigma \curvearrowright x\sigma'$ for all $x \in Dom(\sigma)$, because $x\sigma$ cannot be a proper subterm of $t|_{p_i}$ for any i.

Hence, we have $s = s[l\sigma']_q \to_{U_{cs}(\mathcal{R}) \cup C_E} s' = s[r\sigma']_q$ and $t' = t[r\sigma]_q \curvearrowright s'$, because $s' = C'[s'_1, \ldots s'_m]_{q_1,\ldots,q_m}$ and $t' = C'[t'_1, \ldots t'_m]_{q_1,\ldots,q_m}$ where t'_i is μ-terminating for all $1 \leq i \leq m$ since it is equal to t_j for some $1 \leq j \leq n$. $\square$

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

Proof. $U_{cs}(\mathcal{R}^{C_E}) = U_{cs}(\mathcal{R}) \uplus C_E$ and $\mathcal{R}^{C_E} = \mathcal{R} \uplus C_E$. 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 μ-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-μ-termination of $U_{cs}(\mathcal{R}^{C_E})$ implies non-μ-termination of $U_{cs}(\mathcal{R}^{C_E})$ on original terms, i.e. terms of the original signature of $\mathcal{R}$ (plus $\{G, A\}$), 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 4 there exists an infinite reduction sequence $D : t_0 \to^*_{U_{cs}(\mathcal{R}^{C_E})} t_1 \to^*_{U_{cs}(\mathcal{R}^{C_E})} \ldots$ starting from a term t_0 with a root symbol from $\Sigma \uplus \{G, A\}$, such that each replacing subterm of t_0 is terminating. We will prove the existence of another infinite reduction D' starting at $t'_0 = peval_{\mathcal{R}^{C_E}}(t_0)$, which does not contain any U-symbols. Note that $t_0 = C[t_0^1, \ldots t_0^m]_{p_1,\ldots,p_m} \curvearrowright t'_0 = C[peval(t_0^1), \ldots peval(t_0^m)]_{p_1,\ldots,p_m}$ where C is non-empty because t_0 is not a U-term.

Now to prove by induction that an infinite reduction sequence D' starting at t'_0 can be constructed we show that $t_j \curvearrowright t'_j$ implies $t_{j+k} \curvearrowright t'_{j+k}$ for some $k \geq 1$ with $t'_j \to^+_{U_{cs}(\mathcal{R}^{C_E})} t'_{j+k}$.

Assume $t_j \curvearrowright t'_j$, i.e. $t_j = C[t_j^1, \ldots, t_j^n]_{p_1,\ldots,p_n}$ and $t'_j = C[peval(t_j^1), \ldots, peval(t_j^n)]_{p_1,\ldots,p_n}$. Consider the subreduction $t_j \to_{U_{cs}(\mathcal{R}^{C_E})} t_{j+1} \cdots \to_{U_{cs}(\mathcal{R}^{C_E})} t_{j+k}$ of D such that the last step of this subreduction occurs at a position $q < p_i$ for some $1 \leq i \leq n$. Note that such a reduction must appear in each tail of D, because the terms $t_j^1, \ldots t_j^n$ are all μ-terminating.

We get $t'_j \to^+_{U_{cs}(\mathcal{R}^{C_E})} t'_{j+k}$ and $t_{j+k} \curvearrowright t'_{j+k}$ through iterated (k times) applications of Lemma 5.

Hence, we can construct an infinite $U_{cs}(\mathcal{R}^{C_E})$-reduction sequence starting from t'_0 which implies non-cs-quasi-reductivity of $\mathcal{R}^{C_E}$ according to Corollary 4. □

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

Corollary 5. *The property of C_E-cs-quasi-reductivity is modular for disjoint unions.*

Corollary 6. *C_E-operational termination (defined for a DCTRS $\mathcal{R}$ as operational termination of $\mathcal{R} \uplus C_E$) is modular for disjoint unions.*

Example 9. *Consider the following DCTRS $\mathcal{R}$*

$$\begin{array}{rcl}
\alpha_1 : div(x,y) \to pair(0,x) & \Leftarrow & greater(y,x) \to^* true \\
\alpha_2 : div(x,y) \to pair(s(q),r) & \Leftarrow & leq(y,x) \to^* true, \\
 & & div(x-y,y) \to^* pair(q,r) \\
x - 0 & \to & x \\
0 - y & \to & 0 \\
s(x) - s(y) & \to & x - y \\
greater(s(x), s(y)) & \to & greater(x,y) \\
greater(s(x), 0) & \to & true \\
leq(s(x), s(y)) & \to & leq(x,y) \\
leq(0,x) & \to & true
\end{array}$$

performing a simple division with residual. Transforming the conditional rules

Property of $\boldsymbol{U_{cs}(\mathcal{R})}$	**Implied property of $\boldsymbol{\mathcal{R}}$**	**Proved in**
μ-Termination	Operational termination	Theorem 3 and Corollary 1
Non-μ-termination	Non-(C_E-operational termination)	Theorem 6
μ-Termination on original terms	Operational termination	Theorem 5 and Corollary 1
Non-μ-(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 $\mathcal{R}$.

α_1 and α_2 yields

$$\begin{array}{rcl} div(x,y) & \to & U_1^{\alpha_1}(greater(y,x),x,y) \\ U_1^{\alpha_1}(true,x,y) & \to & pair(0,x) \\ div(x,y) & \to & U_1^{\alpha_2}(leq(y,x),x,y) \\ U_1^{\alpha_2}(true,x,y) & \to & U_2^{\alpha_2}(div(x-y,y),x,y) \\ U_2^{\alpha_2}(pair(q,r),x,y) & \to & pair(s(q),r) \end{array}$$

$U_{cs}(\mathcal{R})$ consists of these rules and the unconditional rules from $\mathcal{R}$. Indeed $U_{cs}(\mathcal{R})$ is non-μ-terminating

$$\begin{array}{rcl} \underline{div(x,0)} & \to & U_1^{\alpha_2}(\underline{leq(0,x)},x,0) \to \underline{U_1^{\alpha_2}(true,x,0)} \\ & \to & U_2^{\alpha_2}(div(\underline{minus(x,0)},0),x,0) \to U_2^{\alpha_2}(\underline{div(x,0)},x,0) \to \dots \end{array}$$

Hence, we deduce non-C_E-operational termination of $\mathcal{R}$ according to Theorem 6 which points to a flaw in the specification of $\mathcal{R}$ allowing division by zero.

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

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 [18]. 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 problems, in order to take into account our intention of proving termination only on restricted sets of terms, by adding an additional component 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 $u_1\sigma \in \mathcal{T}((\Sigma^\sharp \setminus \Sigma) \cup \Sigma', V)$ (more precisely $root(u_1\sigma) \in \Sigma^\#$ and every proper subterm of $u_1\sigma$ is in $\mathcal{T}(\Sigma', V)$). 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 5. *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 [18], 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.

6.1 Narrowing Processors

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. [18]).

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 $s_i\sigma \to^* t_{i+1}\sigma$ is part of a chain and $s_i\sigma$ and $t_{i+1}\sigma$ are not equal (actually we demand that s_i and t_{i+1} are not unifiable) then the rewrite sequence $s_i\sigma \to^* t_{i+1}\sigma$ 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 $t_i \to s_i$ by the set $\{t_i\theta_j \to s_i^j \mid 1 \leq j \leq n\}$ with $s_i^1, \ldots s_i^n$ being all possible (one step, context-sensitive) narrowings of s_i and $\theta_1, \ldots, \theta_n$ being the¡ corresponding mgu's. Theorem 7 below shows that replacing a rule $t_i \to s_i \in DP$ 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 s_i is linear and does not unify with a left-hand side of any rule in DP.

Analogously, a rule $t_i \to s_i$ occurring in a chain can be replaced under the corresponding preconditions by the set $\{t_i^j \to s_i\theta_j \mid 1 \leq j \leq m\}$, where $t_i^1, \ldots t_i^m$ are the (one step, context-sensitive) *backward* narrowings of t_i and $\theta_1, \ldots, \theta_m$ 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 6 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 7 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 6. *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS. Assume that $u_1 \xrightarrow{\epsilon} v_1, u_2 \xrightarrow{\epsilon} v_2 \ldots$ 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 $u_1\sigma$ 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 7. *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS. Assume that $u_1 \xrightarrow{\epsilon} v_1, u_2 \xrightarrow{\epsilon} v_2 \ldots$ 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 $u_1\sigma$ 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 6 and 7 motivate the definition of two dependency pair processors based on the standard narrowing processor.

Definition 12 (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*

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

and $\{v_i^1, \ldots, v_i^n\}$ is the set of all (one-step, context-sensitive) narrowings of v_i with corresponding mgu's $\theta_1, \ldots, \theta_n$, such that all subterms of v_i^k 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 $(DP, U_{cs}(\mathcal{R}), \mu, \Sigma')$ where $DP = (\Sigma^\sharp, S^\sharp)$ and $U_{cs}(\mathcal{R}) = (\Sigma, R)$ provided that $U_{cs}(\mathcal{R})$ is obtained by the transformation of Definition 4 from 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 6 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, let S be the set of substitutions satisfying $u_j\sigma \to^*_{U_{cs}(\mathcal{R})} v_{j+1}\sigma$ for all $\{j > 0 \mid j \neq i\}$, $u_i\sigma \to^*_{U_{cs}(\mathcal{R})} s\sigma$ and $t\sigma \to^*_{U_{cs}(\mathcal{R})} v_{i+1}\sigma$. Moreover, let $\sigma \in S$ be the substitution such that the reduction sequence $t\sigma \to^*_{U_{cs}(\mathcal{R})} v_{i+1}\sigma$ has minimal length (among all substitutions in S).

We take a closer look at the sequence $t\sigma \to^*_{U_{cs}(\mathcal{R})} v_{i+1}\sigma$ 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

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

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

Hence, the sequence $t\sigma \to^*_{U_{cs}(\mathcal{R})} v_{i+1}\sigma$ 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 $v_1 \to u_1, \ldots v_i \to u_i, s\theta \to \bar{t}, v_{i+1} \to u_{i+1}, \ldots$ 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 $t\sigma \to^*_{U_{cs}(\mathcal{R})} v_{i+1}\sigma$ starts with a single reduction $t\sigma = t[l\rho]_p\sigma \to_{U_{cs}(\mathcal{R})} t[r\rho]_p\sigma$ using a rule $l \to r$. Since we consider l and t to be variable disjoint, we extend σ so that $x\sigma = x\rho$ 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 $\bar{t} = t[r\theta]_p$ and since $s\theta \to \bar{t}$ 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 $s\theta$ and $\bar{t}$. Thus,

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

and $v_1 \to u_1, \ldots v_i \to u_i, s\theta \to \bar{t}, v_{i+1} \to u_{i+1}, \ldots$ 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 $(\mathcal{P} \cup \{s\theta_1 \to t_1, \ldots s\theta_n \to t_n\}, \mathcal{R}, \mu, \Sigma')$ be $Proc_{\text{RFN}}(P)$. We show that if $v_1 \to u_1, \ldots, v_i \to u_i, s\theta_m \to t_m, v_{i+1} \to u_{i+1}, \ldots$ is a $(\mathcal{P} \cup \{s\theta_1 \to t_1, \ldots s\theta_n \to$

$t_n\}, \mathcal{R}, \mu)$-chain for some $1 \leq m \leq n$, then $v_1 \to u_1, \ldots, v_i \to u_i, s \to t, v_{i+1} \to u_{i+1}, \ldots$ is a chain as well.

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

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

$$u_i\sigma' \to^*_{U_{cs}(\mathcal{R})} s\theta\sigma = s\sigma'$$

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

$$t\sigma' = t\theta\sigma \to^*_{U_{cs}(\mathcal{R})} t_m\sigma' \to^*_{U_{cs}(\mathcal{R})} v_{i+1}\sigma'$$

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

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 10. *Consider the DP problem $\mathcal{P} = (DP, \mathcal{R}, \mu, \Sigma)$ given by*

$$DP = \begin{cases} t^\#(f(x)) & \to & t^\#(h(x)) \\ t^\#(b) & \to & t^\#(f(a)) \end{cases}$$

$$\mathcal{R} = \begin{cases} a & \to & b \\ h(x) & \to & U(x,x) \\ U(x,x) & \to & x \end{cases}$$

$\Sigma = \{a, b, f, h, t\}$ and $\mu(g) = \{1\}$ for all $g \in \{h, U, t, t^\#\}$, $\mu(g) = \emptyset$ for all $g \in \{f\}$. Note that $\mathcal{R}$ is the transformed version of the DCTRS $\{a \to b, h(x) \to x \Leftarrow x \to^* x\}$. $\mathcal{P}$ is infinite because there exists an infinite DP chain:

$$t^\#(f(a)) \xrightarrow{\epsilon} t^\#(h(a)) \to_\mu t^\#(h(b)) \to_\mu t^\#(U(b,b)) \to_\mu t^\#(b) \xrightarrow{\epsilon} t^\#(f(a))$$

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. Narrowing the first pair and thus replacing it by $t^\#(\underline{f(x)}) \to t^\#(U(x,x))$ would yield a finite DP problem. Thus the precondition $Var^\mu(l) \cap Var^\mu(r) = \emptyset$ for the narrowed rule $l \to r \in DP$ is really needed.

The second dependency pair processor makes use of backward narrowing.

Definition 13 (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*

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

and $\{u_i^1, \dots, u_i^n\}$ is the set of (one-step, context-sensitive) backward *narrowings of u_i with corresponding mgu's $\theta_1, \dots, \theta_n$, such that all subterms of u_i^k are* reachable *from Σ'-terms for all $1 \le k \le n$.*

Theorem 8. *The dependency pair processor $Proc_{\text{RBN}}$ is sound and complete for an SS-CS-DP-problem $(DP, U_{cs}(\mathcal{R}), \mu, \Sigma')$ where $DP = (\Sigma^\sharp, S^\sharp)$ and $U_{cs}(\mathcal{R}) = (\Sigma, R)$ provided that $U_{cs}(\mathcal{R})$ is obtained by the transformation of Definition 4 from 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. 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. This heuristic is also used in the implementation of these processors in VMTL [32]. Note that when using approximations of the notions "reducible to" and "reachable from" the narrowing processors may no longer be complete (cf. Example 11), hence they cannot be used to prove non-termination on original terms in general.

Examples 11 and 12 below show that this simple approximation is already sufficient to prove termination on original terms where ordinary termination does not hold (Example 11), or to significantly reduce the number of narrowings that have to be considered (Example 12).

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 [21]).

Example 11. *Consider the transformed CSRS $\mathcal{R}$ of Example 8 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. $DP(\mathcal{R}) = \{f^\sharp(k(a), k(b), x) \to f^\sharp(x, x, x)\}$[8] . Applying $Proc_{\text{RBN}}$ to $\mathcal{P}_0$, we obtain a new problem $\mathcal{P}_1 = (DP_1, \mathcal{R}, \mu, \Sigma')$ where*

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

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

$Proc_{\text{RBN}}$ 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.

Note that in this example it is critical to discard narrowings that contain the term $U(d,d)$, because this term is not reachable by an original term. If one used to rough approximations for reachability by original terms and considered terms containing $U(d,d)$ as valid terms appearing on DP chains, then indeed infinite DP-chains would exist. However, the conclusion that the system is non-μ-terminating on original terms would be incorrect, because when using approximations for the notion "reachable from" the backward narrowing processor is no longer complete.

Example 12. *Consider the transformed CSRS $\mathcal{R}$ of Example 3. 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

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

$Proc_{\text{RFN}}$ 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[9] *, those narrowings can be disregarded according to Definition 12. 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 tuples) 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 14 below).*

6.2 Instantiation Processors

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 still we need a way to propagate these results in proofs of termination.

To this end we propose an *instantiation* processor, whose informal goal is to propagate the results of condition evaluations approximated through narrowing

[9] A term t is *root stable* w.r.t. to a rewrite system $\mathcal{R}$ if there is no $\mathcal{R}$ reduction issuing from t that contains a root reduction step.

to subsequent conditions (i.e. subsequent rules in the transformed system).[10] The following lemma provides the theoretical basis for our instantiation processor.

Lemma 8. *Let $\mathcal{P} = (\Sigma, R)$ and $\mathcal{R} = (\mathcal{D} \uplus \mathcal{C}, R')$ be TRSs with a combined replacement map μ. If $s\theta \xrightarrow{\epsilon}_{\mathcal{P},\mu} t\theta \xrightarrow{>\epsilon *}_{\mathcal{R},\mu} s'\theta' \xrightarrow{\epsilon}_{\mathcal{P},\mu} t'\theta'$, $s'\sigma = t$ 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. $s'|_p \in Var \Rightarrow \forall q < p\colon root(s'|_q) \in (\Sigma \cup \mathcal{C}) \setminus \mathcal{D}$), then $s'\sigma\theta \xrightarrow{\epsilon}_{\mathcal{P},\mu} t'\sigma\overline{\theta} \to^*_{\mathcal{R},\mu} t'\theta'$ for some $\overline{\theta}$, such that $x\overline{\theta} = x\theta$ for all $x \in Var(t)$.*

Definition 14 (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 antecedents of the pair $s \to t$ on $(DP, \mathcal{R}, \mu)$-chains.*[11] *If, for all $l \to r \in Pred_{s\to t}$,$r = s\sigma$ for some σ, then the processor $Proc_{\mathrm{BI}}$ yields $(DP' \cup \{s\sigma \to t\sigma \mid l \to r \in Pred_{s\to t} \wedge r = s\sigma\}, \mathcal{R}, \mu, \Sigma')$.*

Theorem 9. *The processor $Proc_{\mathrm{BI}}$ 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 $Proc_{\mathrm{BI}}(\mathcal{P}) = \mathcal{P}'$.

Consider an arbitrary fragment of the initial infinite chain:

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

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

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

(according to Lemma 8).

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

[10] Note that our instantiation processor is similar to, but incomparable to the one in [18], as in [18] variables are only instantiated by constructor terms while according to Definition 14 in our approach also terms containing defined symbols can be substituted (cf. Example 13 below).

[11] To be precise this definition of $Pred_{s\to t}$ identifies a superset of potential antecedent pairs of $s \to t$ in DP chains. The exact set is in general undecidable, however one could use other/better approximations here as well.

Example 13. *Consider an SS-CS-DP-problem* $P = (DP, \mathcal{R}, \mu, \Sigma')$ *where*

$$DP = \left\{ \begin{array}{rcl} d^{\#} & \to & U_1^{\#}(c) \\ U_1^{\#}(x) & \to & c^{\#} \end{array} \right.$$

$$\mathcal{R} = \left\{ \begin{array}{rcl} d & \to & U_1(c) \\ U_1(x) & \to & c \\ c & \to & b \end{array} \right.$$

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

$$\begin{array}{rcl} d & \to & x \Leftarrow c \to^* x \\ c & \to & b \end{array}$$

The backward instantiation processor can be applied to P. *The dependency pair* $s \to t$ *is* $U_1^{\#}(x) \to x$ *and its only potential antecedent is* $d^{\#} \to U_1^{\#}(c)$. *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 14 the result of the application of the processor is a new dependency pair problem* $(DP', \mathcal{R}, \mu, \Sigma')$ *where*

$$DP' = \left\{ \begin{array}{rcl} d^{\#} & \to & U_1^{\#}(c) \\ U_1^{\#}(c) & \to & c^{\#} \end{array} \right.$$

Note that finiteness of this resulting SS-CS-DP-problem is obvious and can easily be shown be repeated application of the forward narrowing processor of Definition 12.

Example 14. *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 3 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 15 (forward 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* 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 descendants of the pair* $s \to t$ *on* $(DP, \mathcal{R}, \mu)$*-chains.*[12] *If,*

[12]To be precise this definition of $Succ_{s \to t}$ identifies a superset of potential descendant pairs of $s \to t$ in DP chains. The exact set is in general undecidable, however one could use other/better approximations here as well.

for all $l \to r \in Succ_{s\to t}$, $l = t\sigma$ *for some* σ, *then the processor* $Proc_{\text{FI}}$ *yields* $(DP' \cup \{s\sigma \to t\sigma \mid l \to r \in Succ_{s\to t} \wedge l = t\sigma\}, \mathcal{R}, \mu, \Sigma')$.

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 8.

Lemma 9. *Let* $\mathcal{P} = (\Sigma, R)$ *and* $\mathcal{R} = (\mathcal{D} \uplus \mathcal{C}, R')$ *be TRSs with a combined replacement map* μ. *If* $s\theta \xrightarrow{\epsilon}_{\mathcal{P},\mu} t\theta \xrightarrow{>\epsilon^*}_{\mathcal{R},\mu} s'\theta' \xrightarrow{\epsilon}_{\mathcal{P},\mu} t'\theta'$, $t\sigma = s'$ *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.* $t|_p \in Var \Rightarrow \forall q < p\colon root(t|_q) \in (\Sigma \cup \mathcal{C}) \setminus \mathcal{D}$*), then* $s\theta \to^*_{\mathcal{R},\mu} s\sigma\overline{\theta}$ *for some* $\overline{\theta}$, *such that* $x\overline{\theta} = x\theta'$ *for all* $x \in Var(t\sigma)$.

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:

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

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 $s_{i+1} = t_i\sigma$ and thus there is a dependency pair $s_i\sigma \to t_i\sigma$ in $\mathcal{P}'$. In the latter case the new chain fragment is

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

(according to Lemma 9).

Completeness: Consider an infinite chain w.r.t. $\mathcal{P}'$. $\ldots s_i\sigma\theta \xrightarrow{\epsilon} t_i\sigma\theta \ldots$. As we assume that all dependency pairs in chains are variable disjoint, we can adapt θ to behave like $\sigma\theta$ 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 DCTRSs, 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 DCTRS are satisfiable.

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.

In the experiments we performed to evaluate our approach, the combination of narrowing and instantiation was only part of the strategy for finding proofs in the dependency pair framework. More precisely, we applied the narrowing processors (backward and forward in parallel; cf. [32, Section 3.1]) until they were no longer applicable and used the instantiation processors afterwards. See Section 6.3 for details on other DP processors available in our tool VMTL.

Example 15. *In Example 11, 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.*

6.3 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 ([32]). Moreover, VMTL contains implementations of various standard (mostly ordering based) DP processors. These are

- a dependency graph processor,
- reduction pair processors based on RPOS and polynomial orderings, and
- a size-change-principle processor.

In addition a simple check for infinity of DP problems is included that can be viewed as a DP processor returning "`no`", hence enabling VMTL to prove non-termination. Note that, as the narrowing processors (using approximations for deciding reducibility to resp. reachability from) are not complete, infinity of a DP problem does not imply non-termination of the original rewrite system on original terms after they have been used during the proof search. For a more thorough description of the features of VMTL we refer to [32]. The results and details of our tests can be found at the tool's homepage.[1] Out of 27 tested systems our implementation was able to prove operational termination of 17. Note that for only one DCTRS in this collection the transformed system is not μ-terminating on all, but only on original terms (i.e. Example 8). However, we refrained from providing more examples of this kind, since we conjecture that they would all have a structure similar to the DCTRS in Example 8. This conjecture is supported by the fact that for such TRSs $\mathcal{R}$ (i.e., where $\mathcal{R}$ is operationally terminating while $U_{cs}(\mathcal{R})$ is non-μ-terminating), $\mathcal{R} \cup C_E$ is not operationally terminating (cf. Theorem 6 and Corollary 4). Moreover, DCTRSs with this property are rather pathological and do not arise naturally as program specifications. We showed, however, that our approach is useful also for proving termination of DCTRSs not belonging to this class. This is supported by our experiments where operational termination of several DCTRSs $\mathcal{R}$ could be shown whereas they could not be handled by traditional methods despite $\mathcal{R} \cup C_E$ being operationally terminating as well.

The examples used in the experiments were taken from the termination problem database (TPDB)[13] and from standard literature on conditional term rewriting (e.g. [31] and [26]).

In our experiments other termination tools supporting conditional rewrite systems scored worse on this set of examples. We tested AProVE [17] on the set of examples. It was able to prove operational termination of 15 through the web-interface. However, the batch version (i.e. AProVE 1.2[14]) could only prove operational termination of 12 examples. This illustrates that termination of CSRSs obtained by our transformation may be hard to verify, and sophisticated and complicated proof methods (as implemented only in the most recent version of AProVE) may be needed.

Overall, VMTL was able to prove operational termination of 6 DCTRSs for which AProVE failed. On the other hand, operational termination of 4 other DCTRSs could only be successfully proved by AProVE. In the 6 examples where VMTL was successful while AProVE was not, the narrowing and instantiation processors of Section 6 played a crucial rule.

On the negative side, repeated application especially of narrowing processors can be expensive with respect to execution time (and space). Yet, we did not restrict the application of the narrowing and instantiation processors by imposing complex applicability conditions as for instance described in [18, Section 5.2] using the concept of *safe transformations*. The reason is that for the particular class of rewrite systems obtained by transformations from conditional systems it might be necessary to spend more time on narrowing and instantiation techniques than on the search for applicable orderings. Still such applicability conditions tailored to systems obtained by the transformation of Definition 4 would be an interesting direction for future work.

Note also that 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.

7 Related Work

The idea of using context-sensitivity to improve the unraveling transformation of [26, 27, 30, 31] is not new. In [10, 29, 11] the same idea is used in conjunction with another optimization. The second optimization is to store the bindings of

[13] `http://www.lri.fr/~marche/tpdb/`

[14] Newer batch versions of AProVE failed to prove termination of any DCTRSs with extra variables in our experiments.

only those variables in the arguments of a U_j^α symbol that occur in a subsequent condition or in the right-hand side of the rule α.

For clarity we provide a formal definition of this optimization.

Definition 16 (Optimized transformation according to [10, 29, 11])**.** *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:*

$$\begin{array}{rcl} l & \to & U_1^\alpha(s_1, \vec{x}_1) \\ U_1^\alpha(t_1, \vec{x}_1) & \to & U_2^\alpha(s_2, \vec{x}_2) \\ & \vdots & \\ U_n^\alpha(t_n, \vec{x}_n) & \to & r \end{array}$$

Here the sequences of variables $\vec{x}_i$ are given by (an arbitrary but fixed sequentialization of the set of variables)

$$(\mathit{Var}(l) \cup \mathit{Var}(t_1) \ldots \mathit{Var}(t_{i-1})) \cap$$
$$(\mathit{Var}(t_i) \cup \mathit{Var}(s_{i+1}) \cup \mathit{Var}(t_{i+1}) \ldots \mathit{Var}(s_n) \cup \mathit{Var}(t_n) \cup \mathit{Var}(r)).$$

The transformed system $U_{opt}(\mathcal{R}) = (U(\Sigma), U_{opt}(R))$ is obtained by transforming each rule of $\mathcal{R}$ where $U(\Sigma)$ is Σ extended by all new function symbols. We use a replacement map μ_{opt} given by $\mu_{opt}(U) = \{1\}$ for every auxiliary symbol U (i.e. $U \in U(\Sigma) \setminus \Sigma$) and $\mu_{opt}(f) = \{1, \ldots, ar(f)\}$ for every $f \in \Sigma$.

Indeed, according to [11] it holds that whenever $U_{opt}(\mathcal{R})$ is μ_{opt}-terminating, $\mathcal{R}$ is operationally terminating.[15] [16] Since the transformation of Definition 16 produces smaller transformed systems than the one from Definition 4, it might be advantageous to use it in termination analysis. However, there is a price to pay for this optimization. That is, one loses the property of simulation-soundness (cf. Theorem 2).

Example 16. *Consider a DCTRS $\mathcal{R}$ given by*

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

The transformed system $U_{cs}(\mathcal{R})$ consists of the following rules

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

[15]The transformation we presented in Definition 16 is actually a special case of the transformation introduced in [11]. There, the authors work in a more general setting where $\mathcal{R}$ itself may be context-sensitive and rewriting modulo an equational theory is used.

[16]Note, however, that in [11, p. 78] the authors introduce both $U_{cs}(\mathcal{R})$ and $U_{opt}(\mathcal{R})$, but do not clearly distinguish between them subsequently. This appears to be justified in the context of [11, Theorem 2 and Lemma 3] (because the proofs of these latter results work for both versions of the transformation), but not in general, since the two transformations have different properties, cf. Examples 16, 17! In particular, μ_{opt}-termination of $U_{opt}(\mathcal{R})$ implies $\mu_{U_{cs}(\mathcal{R})}$-termination of $U_{cs}(\mathcal{R})$, but not vice versa.

$U_{cs}(\mathcal{R})$ is μ-terminating and thus $\mathcal{R}$ is operationally terminating. However, $U_{opt}(\mathcal{R})$ given by

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

is easily seen to be non-(μ_{opt}-)terminating (even on original terms) due to the cyclic reduction sequence

$$g(f(a),f(b)) \to^{+}_{U_{opt}(\mathcal{R})} g(U(a),U(a)) \to_{U_{opt}(\mathcal{R})} g(f(a),f(b))$$

Hence, Theorems 2 and 4 and Corollary 4 do not hold for this optimized transformation.[17]

Note that the DCTRS in Example 16 is not left-linear. However, this property is not crucial as the following left-linear example shows (also the left-hand sides of conditions are linear).

Example 17. *Consider the DCTRS $\mathcal{R}$ given by*

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

$U_{opt}(\mathcal{R})$ is given by

$$\begin{array}{rcl} f(x) & \to & U_f(a) \\ U_f(h(y)) & \to & y \\ g(x,b) & \to & U_g^1(f(b),x) \\ U_g^1(x,x) & \to & U_g^2(x,x) \\ U_g^2(c,x) & \to & g(f(c),x) \\ a & \to & h(b) \\ a & \to & h(c) \end{array}$$

Then $\mathcal{R}$ is operationally terminating (however, this has been proved by hand – via analyzing the shape of potentially existing minimal counterexamples – as automated termination tools currently fail to prove μ-termination of $U_{cs}(\mathcal{R})$), but

[17] Technically, this is reflected in the fact that the definition of the back-translation function tb (which is crucial for the proofs of these results) according to Definition 3 would not be well-defined for $U_{opt}(\mathcal{R})$ instead of $U_{cs}(\mathcal{R})$! The reason is that the substitution σ involved in Definition 3 may become non-well-defined due to the existence of erasing rules in $U_{opt}(\mathcal{R})$ that forget certain variable bindings (cf. the first rule of $U_{opt}(\mathcal{R})$ in Examples 16 and 17, respectively). For a more general and thorough discussion of (desirable) properties of back-translation functions in the setting of transforming CTRSs into TRSs we refer to [19].

$U_{opt}(\mathcal{R})$ is again non-terminating due to the following cyclic reduction sequence.

$$\begin{array}{rcl} g(f(c),b) & \to & g(U_f(a),b) \to U_g^1(f(b),U_f(a)) \to U_g^1(U_f(a),U_f(a)) \\ & \to & U_g^2(U_f(a),U_f(a)) \to U_g^2(U_f(h(c)),U_f(a)) \to U_g^2(c,U_f(a)) \\ & \to & g(f(c),U_f(a)) \to g(f(c),U_f(h(b))) \to g(f(c),b) \end{array}$$

In [29] it is shown that left-linearity (and right linearity in combination with non-erasingness) of the transformed system $U_{opt}(\mathcal{R})$ is sufficient to guarantee simulation-soundness (even if context-sensitivity is dropped).

However, despite being an interesting question we refrain from giving a more precise assessment of conditions under which the optimized transformation is simulation sound. Yet, solving this problem could also be useful in practice because automated termination provers could base the decision on which transformation to use on this knowledge.

In [29] simulation-soundness is obtained by restricting $U_{opt}(\mathcal{R})$-evaluations.[18] The idea is to contract only redexes not containing auxiliary U-symbols. Hence, it would be sufficient to prove termination of U_{opt} under this restriction in order to deduce operational termination of the original conditional system. While this might be feasible, given the recent advances in proving termination under strategies (cf. e.g. [12]), no concrete methods for this particular task exist to the authors' knowledge.

8 Discussion and Conclusion

We analyzed the context-sensitive modification of the unraveling transformation of DCTRSs into TRSs ([26, 27, 30, 31]). This transformation plays a crucial role in several approaches for the termination analysis of current programming and specification languages (cf., [25, 11]). 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 operational termination (cf. also [16]). 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.), with few exceptions like e.g. [14, 33, 13]. 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.

[18] Note that our notion of simulation-soundness is called simulation-completeness there.

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 [29] and [28] 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 our context-sensitive transformation is simulation sound and simulation complete in their sense.

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 11).

9 Acknowledgments

We would like to thank the anonymous referees for numerous useful hints and remarks and particularly for pointing out a flaw in the proof of Theorem 2 in a previous version of this paper.

References

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

[2] B. Alarcón, R. Gutiérrez, J. Iborra, and S. Lucas. Proving termination of context-sensitive rewriting with MU-TERM. In P. Lucio and F. Orejas, eds., *Proc. 6th Spanish Conference on Programming and Languages (PROLE'06)*, *Electronic Notes in Theoretical Computer Science*, 188:105–115, 2007.

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

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

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

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

[7] M. Bezem, J. Klop, and R. de Vrijer, eds. *Term Rewriting Systems.* Cambridge Tracts in Theoretical Computer Science 55. Cambridge University Press, Mar. 2003.

[8] 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.

[9] 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 Int. Conf. on Rewriting Techniques and Applications (RTA'03), Valencia, Spain*, LNCS 2706, pp. 76–87, Springer-Verlag, 2003.

[10] 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*, pp. 147–158, ACM Press, 2004.

[11] 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.

[12] J. Endrullis and D. Hendriks. From outermost to context-sensitive rewriting. In R. Treinen, ed., *Proc. 20th Int. Conf. on Rewriting Techniques and Applications (RTA'09), Brasília, Brazil*, LNCS 5595, pp. 305–319, Springer-Verlag, 2009.

[13] J. Endrullis, R. de Vrijer and J. Waldmann. Local Termination. In R. Treinen, ed., *Proc. 20th Int. Conf. on Rewriting Techniques and Applications (RTA'09), Brasília, Brazil*, LNCS 5595, pp. 270–284, Springer-Verlag, 2009.

[14] O. Fissore, I. Gnaedig and H.Kirchner. CARIBOO: A multi-strategy termination proof tool based on induction. In A. Rubio, ed., *Proc. 6th Int. Workshop on Termination (WST'03), Valencia, Spain*, Technical Report DSIC-II-15/03, UPV, pp. 77–79. 2003.

[15] 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 Int. Workshop on Conditional Term Rewriting Systems (CTRS'92), Pont-à-Mousson, France*, LNCS 656, pp. 430–437. Springer-Verlag, 1993.

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

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

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

[19] 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 5486, pp. 166–186. Springer-Verlag, 2009.

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

[21] 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), Jerusalem, Israel*, LNCS 968, pp. 247–261. Springer-Verlag, 1995.

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

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

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

[25] S. Lucas and J. Meseguer. Operational termination of membership equational programs: The order-sorted way. In G. Roşu, ed., *Proc. 7th Int. Workshop on Rewriting Logic and its Applications (WRLA'08)*, Electronic Notes in Theoretical Computer Science, 238(3):207–225, 2009.

[26] 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*, LNCS 1139, pp. 107–121. Springer-Verlag, 1996.

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

[28] N. Nishida, T. Mizutani, and M. Sakai. Transformation for refining unraveled conditional term rewriting systems. *Electronic Notes in Theoretical Computer Science*, 174(10):75–95, 2007.

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

[30] 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-Verlag, 1999.

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

[32] 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), Brasília, Brazil*, LNCS 5595, pp. 285–294, Springer-Verlag, 2009.

[33] P. Schneider-Kamp, J. Giesl, A. Serebrenik and R. Thiemann. Automated termination analysis for logic programs by term rewriting. In G. Puebla, ed., *Proc. 16th Int. Symp. on Logic-Based Program Synthesis and Transformation (LOPSTR'06), Venice, Italy*, LNCS 4407, pp. 177-193. Springer-Verlag, 2007.

[34] T.-F. Şerbănuţă and G. Roşu. Computationally equivalent elimination of conditions. In F. Pfenning, ed., *Proc. 8th International Conference on Rewriting Techniques and Applications (RTA'06)*, LNCS 4098, pp. 19–34. Springer-Verlag, 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., $s|_p = l\sigma$). From the applicability of the conditional rule it follows that σ can be extended to σ' such that $s_i\sigma' \rightarrow^*_\mathcal{R} t_i\sigma'$ 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 $s_i\sigma' \succeq t_i\sigma'$ for all $1 \leq i \leq n$. Hence, by cs-quasi-reductivity we get $l\sigma' \succ_\mu r\sigma'$, 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 $> := \succ^{st}_\mu |_{\mathcal{T}(\Sigma,V)\times\mathcal{T}(\Sigma,V)}$:

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 $t \in \mathcal{T}(\Sigma, V)$ $(t = C[s]_p)$, such that $t \not> s$. This implies $t \not\succ^{st}_\mu s$, which contradicts the definition of $\succ^{st}_\mu$ 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 $s_j\sigma \rightarrow^* t_j\sigma$ for every $j \in \{1, \ldots, i\}$ implies $l\sigma > s_{i+1}\sigma$. We know that $s_j\sigma \rightarrow^* t_j\sigma \Rightarrow s_j\sigma \succeq_\mu t_j\sigma$. Because of cs-quasi-reductivity this implies $l\sigma \succ^{st}_\mu s_{j+1}\sigma$ and thus $l\sigma > s_{j+1}\sigma$, since $l\sigma, s_{j+1}\sigma \in \mathcal{T}(\Sigma, V)$.

$\square$

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

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

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

□

Proposition 4 *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 4 *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 [24]. Theorem 5, Proposition 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 5 *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 [20], 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 5 *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 $t_1^\sharp = l^\sharp\sigma$ 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 $l\sigma$. If $l^\sharp\sigma \in \mathcal{T}((\Sigma^\sharp \setminus \Sigma) \cup \Sigma', V)$ then $l\sigma \in \mathcal{T}(\Sigma', V)$. 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]. $\square$

Lemma 6 *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS. Assume that $u_1 \xrightarrow{\epsilon} v_1, u_2 \xrightarrow{\epsilon} v_2 \dots$ 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 $u_1\sigma$ 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. According to Corollary 2 we get for all $i \geq 1$ that $v_i\sigma \to^*_{U_{cs}(\mathcal{R})} u_{i+1}\sigma$ implies $\mathsf{tb}(v_i\sigma) \to^*_{U_{cs}(\mathcal{R})} \mathsf{tb}(u_{i+1}\sigma)$ where $\mathsf{tb}(v_i\sigma) = v_i\sigma'$, resp. $\mathsf{tb}(u_{i+1}) = u_{i+1}\sigma'$ and $x\sigma' = \mathsf{tb}(x\sigma)$ for all $x \in Dom(\sigma)$ because v_j, u_j do not contain U-symbols (only $U^\sharp$-symbols) for all $j \geq 1$.

Note that Corollary 2 is applicable because $u_1\sigma$ is an original term (dependency pair symbols are interpreted as constructors when applying Corollary 2) and thus $v_1\sigma$ is an original term as well, thus $v_1\sigma \to^* u_2\sigma$ implies $tb(v_1\sigma) \to tb(u_2\sigma) = u_2\sigma'$ where $x\sigma' = tb(x\sigma)$ for all $x \in Dom(\sigma)$ and moreover $x\sigma' \to^* x\sigma$. Thus $tb(v_2\sigma) = v_2\sigma' \to^* v_2\sigma$, i.e. $v_2\sigma$ is reachable by an original term and thus Corollary 2 is applicable to the reduction sequence $v_2\sigma \to^* u_3\sigma$. Analogously, reachability of $v_i\sigma$ by an original term can be shown for all $i > 0$ and thus the application of Corollary 2 is justified.

It remains to be shown that it is not necessary to introduce a U-term u, such that no descendant of u has a root symbol from Σ in any of the reductions $D_i : v_i\sigma' \to^*_{U_{cs}(\mathcal{R})} u_{i+1}\sigma'$. In order to show this we identify those reduction steps in D_i where the reductum is such a U-term or is inside such a U-term and call them *U-steps*.

Assume D_i contains a reduction sequence $s \overset{p1,\dots,p_n}{\to}{}^*_{U_{cs}(\mathcal{R})} s' \to_{U_{cs}(\mathcal{R})} s''$ such that the first steps are U-steps and the last one is not. Then the last step, say using a rule $l \to r$, occurs outside all U-subterms of s' that are not reduced to original terms by properly finishing the simulated conditional rule application in D_i. Hence, the first steps occur in the variable part of (or parallel to) the second one (because U-symbols occur only at but not below the root position of rules in $U_{cs}(\mathcal{R})$) and we can change the order of the steps and perform the last step first, i.e., $s \to_{U_{cs}(\mathcal{R})} \overline{s}$ and the U-steps afterwards. Depending on whether the variables of l to which superterms of $s|_{p_i}$ are bound in the reduction, are eliminated, copied or duplicated, zero, one or several U-reductions are necessary to derive s'', s.t. $\overline{s} \to^*_{U_{cs}(\mathcal{R})} s''$. Note that non-linearity of l is not a problem if we perform these rearrangements always for the first U-step(s) occurring in the reductions $v_i\sigma' \to^*_{U_{cs}(\mathcal{R})} u_{i+1}\sigma'$, because whenever two U-terms, that have been introduced in a $U_{cs}(\mathcal{R})$-reduction are equal, the tb version of these terms are equal as well. Thus, if all U-steps preceding the non-U-steps are shifted after this step simultaneously, this is also possible in the presence of non-left-linear rules.

Hence, we can shift all U-steps to the end of the reduction sequence D_i. However, as $u_{i+1}\sigma'$ is an original term and the reductum of each U-step is per definition inside a U-term the number of U-steps at the end of D_i must be zero after this rearrangement and thus $v_i\sigma' \to^*_{U_{cs}(\mathcal{R})} u_{i+1}\sigma'$ is without U-steps. $\square$

Lemma 7 *Let $\mathcal{R} = (\Sigma, R)$ be a DCTRS. Assume that $u_1 \xrightarrow{\epsilon} v_1, u_2 \xrightarrow{\epsilon} v_2 \dots$ 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 $u_1\sigma$ 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. According to Corollary 2 we get for all $i \geq 1$ that $v_i\sigma \to^*_{U_{cs}(\mathcal{R})} u_{i+1}\sigma$ implies $\mathsf{tb}(v_i\sigma) \to^*_{U_{cs}(\mathcal{R})} \mathsf{tb}(u_{i+1}\sigma)$ where $\mathsf{tb}(v_i\sigma) = v_i\sigma'$, resp. $\mathsf{tb}(u_{i+1}) = u_{i+1}\sigma'$ and $x\sigma' = \mathsf{tb}(x\sigma)$ for all $x \in Dom(\sigma)$ because v_j, u_j do not contain U-symbols (only $U^\sharp$-symbols) for all $j \geq 1$.

Note that Corollary 2 is applicable because $u_1\sigma$ is an original term (dependency pair symbols are interpreted as constructors when applying Corollary 2) and thus $v_1\sigma$ is an original term as well, thus $v_1\sigma \to^* u_2\sigma$ implies $tb(v_1\sigma) \to tb(u_2\sigma) = u_2\sigma'$ where $x\sigma' = tb(x\sigma)$ for all $x \in Dom(\sigma)$ and moreover $x\sigma' \to^* x\sigma$. Thus $tb(v_2\sigma) = v_2\sigma' \to^* v_2\sigma$, i.e. $v_2\sigma$ is reachable by an original term and thus Corollary 2 is applicable to the reduction sequence $v_2\sigma \to^* u_3\sigma$. Analogously, reachability of $v_i\sigma$ by an original term can be shown for all $i > 0$ and thus the application of Corollary 2 is justified.

Hence, whenever a term s not reachable from an original term occurs in the dependency pair chain, then we have $v_i\sigma' \to^*_{U_{cs}(\mathcal{R})} C[s]$. However, according to Lemma 2 this implies that s is reachable from an original term contradicting the existence of such an s. Note that the root symbol of v_i is interpreted as a constructor when applying Lemma 2. $\square$

Lemma 8 *Let $\mathcal{P} = (\Sigma, R)$ and $\mathcal{R} = (\mathcal{D} \uplus \mathcal{C}, R')$ be TRSs with a combined replacement map μ. If $s\theta \xrightarrow{\epsilon}_{\mathcal{P},\mu} t\theta \xrightarrow{>\epsilon^*}_{\mathcal{R},\mu} s'\theta' \xrightarrow{\epsilon}_{\mathcal{P},\mu} t'\theta'$, $s'\sigma = t$ 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. $s'|_p \in Var \Rightarrow \forall q < p \colon s'|_q \in (\Sigma \cup \mathcal{C}) \setminus \mathcal{D}$), then $s'\sigma\theta \xrightarrow{\epsilon}_{\mathcal{P},\mu} t'\sigma\overline{\theta} \to^*_{\mathcal{R},\mu} t'\theta'$ for some $\overline{\theta}$, such that $x\overline{\theta} = x\theta$ 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 $x_i\sigma\theta \to^*_{\mathcal{R},\mu} x_i\theta'$, as $x_i\sigma\theta = t|_q\theta$ and $x_i\theta' = s'|_q\theta'$ and all positions above q are constructors in t and s'. Thus, we set $y\overline{\theta} = y\theta$ for all $y \in \mathit{Var}(\mathit{Codom}(\sigma))$ and obtain $t'|_{q'}\sigma\overline{\theta} \to^*_{\mathcal{R},\mu} t'|_{q'}\theta'$ for any position q' with $t'|_{q'} = x_i$. Note that if q is replacing in s', then so is q' in t'. Otherwise, $x_i\sigma\theta = x_i\theta'$.

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

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

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

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 $x_i\theta \to^*_{\mathcal{R},\mu} x_i\sigma\theta'$, as $x_i\theta = t|_q\theta$, $x_i\sigma\theta' = s'|_q\theta'$ and all positions above q are constructors in t and s'. Thus, we set $y\overline{\theta} = y\theta'$ for all $y \in \mathit{Var}(\mathit{Codomain}(\sigma))$ and obtain $s|_{q'}\theta \to^*_{\mathcal{R},\mu} s|_{q'}\sigma\overline{\theta}$ for any position q' with $s|_{q'} = x_i$. Note that if q is replacing in t, then so is q' in s. Otherwise, $x_i\theta = x_i\sigma\theta'$.

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