Signature Extensions Preserve Termination
An Alternative Proof via Dependency Pairs

Christian Sternagel* and René Thiemann

University of Innsbruck
{christian.sternagel,rene.thiemann}@uibk.ac.at

Abstract. We give the first mechanized proof of the fact that for showing termination of a term rewrite system, we may restrict to well-formed terms using just the function symbols actually occurring in the rules of the system. Or equivalently, termination of a term rewrite system is preserved under signature extensions. We did not directly formalize the existing proofs for this well-known result, but developed a new and more elegant proof by reusing facts about dependency pairs.

We also investigate signature extensions for termination proofs that use dependency pairs. Here, we were able to develop counterexamples which demonstrate that signature extensions are unsound in general. We further give two conditions where signature extensions are still possible.

1 Introduction

Our main objective is to formally show that the termination behavior of (first-order) term rewrite systems (TRSs) [2] does not change under signature extensions. This is an important part of a bigger development inside IsaFoR (an **Isa**belle **Fo**rmalization of **R**ewriting) which is used to generate CeTA (a tool for **Ce**rtified **T**ermination **A**nalysis) [10].[1] All our results have been formalized and machine-checked in the interactive proof assistant Isabelle/HOL [6]. In the following, whenever we speak about *formalizing* something, we mean a machine-checked formalization using Isabelle/HOL.

In the literature, termination of $\mathcal{R}$ (denoted by $\mathsf{SN}(\mathcal{R})$), is usually only defined for terms that do exclusively incorporate function symbols from the *signature* $\mathcal{F}$ of $\mathcal{R}$. Often, it is implicitly assumed that this is equivalent to termination for terms over arbitrary extensions $\mathcal{F}' \supseteq \mathcal{F}$. This is legitimate, since it has been shown that termination is modular under certain conditions (see [5,7] for details) and signature extensions satisfy these conditions. A property P is called *modular*, whenever $P\ \mathcal{R}$ and $P\ \mathcal{S}$, for TRSs $\mathcal{R}$ and $\mathcal{S}$ over disjoint signatures $\mathcal{F}$ and $\mathcal{G}$, implies $P\ (\mathcal{R} \cup \mathcal{S})$. (Note that $P\ x$ is Isabelle/HOL's way of writing a function or predicate P applied to an argument x.) Now, to use modularity of termination to achieve $\mathsf{SN}(\mathcal{R})$ over the signature $\mathcal{F}' \supseteq \mathcal{F}$, we choose $\mathcal{S} = \varnothing$ and $\mathcal{G} = \mathcal{F}' - \mathcal{F}$. Then, the above mentioned conditions are trivially satisfied and we

* This research is supported by FWF (Austrian Science Fund) project P18763.

[1] http://cl-informatik.uibk.ac.at/software/ceta

A. Dawar and H. Veith (Eds.): CSL 2010, LNCS 6247, pp. 514–528, 2010.

obtain $\mathsf{SN}(\mathcal{R}) \Longrightarrow \mathsf{SN}(\mathcal{R} \cup \mathcal{S})$, where the latter system has the same rules as $\mathcal{R}$, but the signature $\mathcal{F}'$.

In this way, the two aforementioned proofs (which both use rather similar proof techniques), can be used to obtain termination preservation under signature extensions. However, the first proof [5] is quite long and complicated even on paper (10 pages, neglecting preliminaries). Concerning the second proof [7]—although short on paper—there are two reasons for not going that way:

1. This proof would require to formalize concepts that are currently not available in our library but are assumed as preliminaries in the paper proof (which is the only reason that the proof is short). This includes, e.g., multi-hole contexts, and functions like *rank*, *top*, etc. Furthermore, some of those concepts seem bulky to formalize, e.g., multi-hole contexts would require that a context having n holes is always applied to exactly n terms. This cannot be guaranteed on the type level without having dependent types and would lead to side-conditions that had to be added to all proofs using multi-hole contexts.
2. We do already have a formalization of many term rewriting related concepts. Thus, it seems only natural to build on top of those available results.

Hence, we take a different road (that may seem as a detour in the beginning). We use $\mathcal{F}(\mathcal{R})$ to denote the signature just containing function symbols that do actually occur in some rule of $\mathcal{R}$. By $\Rrightarrow_{\mathcal{R}}$ we denote the rewrite relation induced by $\mathcal{R}$ just using terms over $\mathcal{F}(\mathcal{R})$ and by $\rightarrow_{\mathcal{R}}$ the same relation but for terms over arbitrary extensions of $\mathcal{F}(\mathcal{R})$ (i.e., $\Rrightarrow_{\mathcal{R}}$ is a restriction of $\rightarrow_{\mathcal{R}}$). Hence, our first main result can be written as

Theorem 1. $\mathsf{SN}(\Rrightarrow_{\mathcal{R}}) \longleftrightarrow \mathsf{SN}(\rightarrow_{\mathcal{R}})$

In the proof, we concentrate on the direction from left to right, since the converse trivially holds. Before we give our general proof idea, we want to show why “the direct approach” is difficult. By “direct” we mean:

> *Assume that there is an infinite sequence in $\rightarrow_{\mathcal{R}}$ and construct an infinite sequence in $\Rrightarrow_{\mathcal{R}}$ out of it.*

For this purpose we would have to provide a function f such that $f\ s \Rrightarrow_{\mathcal{R}} f\ t$ is implied by $s \rightarrow_{\mathcal{R}} t$ for arbitrary terms s and t. This requires that f somehow removes all function symbols that are not in $\mathcal{F}(\mathcal{R})$ from its argument but still preserves any redexes (i.e., subterms where rules are applicable). For example the simple idea to *clean* terms by replacing all subterms $f(\ldots)$ where $f \notin \mathcal{F}(\mathcal{R})$ by the same variable, does not work. The reason is that a given infinite $\rightarrow_{\mathcal{R}}$-derivation might take place strictly below a symbol $f \notin \mathcal{F}(\mathcal{R})$ and then, after turning $f(\ldots)$ into a variable, those reductions can no longer be simulated. To cut a long story short, we stopped at some point to investigate this direction further, since all our approaches became awfully complicated (especially for mechanizing the proof).

Our salvation appeared in the form of *dependency pairs* (DPs). By redirecting the course of our proof into the DP setting [1] and back again, we were able to give a short and (in our opinion) elegant proof of Theorem 1, using the simple technique of cleaning. The reason is that by using DPs we obtain a derivation which contains infinitely many reductions at the root position. And all these root reductions are still possible after cleaning. Note that this also shows that signature extensions are sound for termination problems in the DP setting (Lemma 8)—our second main result.

However, after trying to extend our proof to the DP setting with *minimal chains*, we discovered a counterexample demonstrating that signature extensions are unsound for non-left-linear TRSs. A small modification of this counterexample also shows that the technique of root-labeling [8] in the DP setting with minimal chains—which relies on signature extensions—is also only sound for left-linear TRSs. This refutes the corresponding result in [8] which does not demand left-linearity. (As the modularity results of [5,7] do not consider minimal chains, these results are not affected by our counterexample.)

In total, in this paper we show that signature extensions are possible for termination of TRSs and that they can be used in the DP setting for left-linear TRSs or for non-minimal chains. We also show that the soundness proofs of root-labeling can be repaired by additionally demanding left-linearity.

The structure of our discourse is as follows: In Section 2 we recall some necessary definitions of term rewriting (as used in our formalization). Afterwards, in Section 3, we give some results on DPs. Two of our main results are given in Section 4, where we also formally prove completeness of DPs. Then, in Section 5 we show some applications—including root-labeling—and limitations of our results. Here, we also discuss the problem of signature extensions in combination with minimal chains and show that there is no problem in the left-linear case. We finally conclude in Section 6.

Since all facts we are using have been machine-checked, we do not give any proofs for results from Sections 2 and 3 and refer the interested reader to the IsaFoR sources (freely available from its website). Our formalization of Theorem 1 can be found under the name `SN_wfrstep_SN_rstep_conv` in the theory `DpFramework`. Also in Section 4 we try to skip technical details and give a high-level overview of our proofs.

2 Preliminaries

In IsaFoR we are concerned with first-order *terms* defined by the data type:

datatype (α, β) *term* = *Var* β | *Fun* α $((\alpha, \beta)$ *term list*$)$

Hence, a term is either a *variable*, or a *function symbol* applied to a list of argument terms. Note that this definition does not incorporate any well-formedness conditions. In particular, there is no signature that terms are restricted to. We identify a function symbol by its representation together with its arity. Hence, the function symbol f in the term *Fun* f [] is different from the function symbol f in the term *Fun* f [*Var* x] (the former has arity 0 and the latter arity 1). To increase readability we write terms like the previous two as f (a constant without arguments) and $f(x)$, respectively. A *(rewrite) rule* is a pair of terms and a *TRS* is a set of rules. A TRS is *well-formed* iff all left-hand sides of rules are non-variable terms and for each rule every variable occurring in the right-hand side also occurs in the left-hand side. We write $\mathsf{wf\_trs}\ \mathcal{R}$ to indicate that the TRS $\mathcal{R}$ is well-formed.

Example 2. The TRS $\{\,\mathsf{add}(\mathsf{0}, y) \to y, \mathsf{add}(\mathsf{s}(x), y) \to \mathsf{s}(\mathsf{add}(x, y))\,\}$, encoding addition on Peano numbers, is well-formed.

The *rewrite relation induced by a TRS* $\mathcal{R}$ is obtained by closing $\mathcal{R}$ under substitutions and contexts, i.e., $\to_{\mathcal{R}}$ is defined inductively by the rules:

$$\frac{(l,\, r) \in \mathcal{R}}{l \to_{\mathcal{R}} r} \qquad \frac{s \to_{\mathcal{R}} t}{s\sigma \to_{\mathcal{R}} t\sigma} \qquad \frac{s \to_{\mathcal{R}} t}{C[s] \to_{\mathcal{R}} C[t]}$$

Here, $t\sigma$ denotes the application of a substitution σ to a term t and $C[t]$ denotes substituting the hole in the context C by the term t. Whenever $s \to_{\mathcal{R}} t$, we say that s rewrites (in one step) to t.

A TRS is *terminating/strongly normalizing* iff the rewrite relation $\to_{\mathcal{R}}$ induced by $\mathcal{R}$ is well-founded—denoted by $\mathsf{SN}(\to_{\mathcal{R}})$. (We sometimes write $\mathsf{SN}(\mathcal{R})$ instead of $\mathsf{SN}(\to_{\mathcal{R}})$ to stress that termination is a property depending merely on $\mathcal{R}$.) Termination of a specific term is written as $\mathsf{SN}_{\mathcal{R}}(t)$, i.e., there is no infinite $\to_{\mathcal{R}}$-derivation starting from t.

Using the definition of $\to_{\mathcal{R}}$, termination is formalized as $\mathsf{SN}(\to_{\mathcal{R}}) \equiv \nexists\, \mathbf{t}.\ \forall\, i.\ \mathbf{t}_i \to_{\mathcal{R}} \mathbf{t}_{i\,+\,1}$. Here, we use functions from natural numbers to some type τ, to encode infinite sequences over elements of type τ, which are written by $\mathbf{t}$ in contrast to terms t. We use subscripts to indicate positions in such infinite sequences, i.e., we write $\mathbf{t}_i$ to denote the i-th element in the infinite sequence $\mathbf{t}$.

Remember that by $\mathcal{F}(\mathcal{R})$ we denote the signature of function symbols actually occurring in some rule of $\mathcal{R}$. Using the function

$$\begin{aligned} \mathcal{F}(x) &= \varnothing, \\ \mathcal{F}(f(\vec{ts})) &= \{(f,\, |\vec{ts}|)\} \cup \bigcup \{\mathcal{F}(t) \mid t \in \vec{ts}\}. \end{aligned}$$

$\mathcal{F}(\mathcal{R})$ is obtained by extending $\mathcal{F}(\cdot)$ to TRSs in the obvious way.

Example 3. The signature of the TRS from Example 2 is $\{(\mathsf{add}, 2), (\mathsf{s}, 1), (\mathsf{0}, 0)\}$.

3 Dependency Pairs

To get hold of the *(recursive) function calls* in a TRS, the so called *dependency pairs* are used [1,3].

Definition 4. *The DPs of a TRS $\mathcal{R}$ are defined by*

$$\mathsf{DP}(\mathcal{R}) = \{(l^\sharp,\, f^\sharp(\vec{ts})) \mid \exists\, r.\ (l,\, r) \in \mathcal{R} \wedge f \in \mathcal{D}(\mathcal{R}) \wedge r \trianglerighteq f(\vec{ts}) \wedge l \ntriangleright f(\vec{ts})\}$$

where ($\triangleright$) $\trianglerighteq$ denotes the (proper) subterm relation on terms and $\mathcal{D}(\mathcal{R})$ is the set of defined function symbols in $\mathcal{R}$.[2] *By $\cdot^\sharp$ we denote the operation of marking the root symbol of a term with the special marker $\sharp$. In examples we use capitalization and hence write F instead of $f^\sharp$.*

[2] A function symbol is defined in a TRS, if it occurs as the root of a left-hand side.

Example 5. Since the TRS of Example 2 contains just one "recursive call," we get the single DP $\mathsf{ADD}(\mathsf{s}(x), y) \to \mathsf{ADD}(x, y)$.

Note how DPs get rid of context information. This is exactly what makes them so useful in our proof.

Having DPs, we can use an alternative characterization of nontermination using DP problems and chains. A *DP problem* $(\mathcal{P}, \mathcal{R})$ just consists of two TRSs $\mathcal{P}$ and $\mathcal{R}$. Then a $(\mathcal{P}, \mathcal{R})$-*chain* is an infinite sequence of the following shape:

$$\forall\, i.\ (\mathbf{s}_i,\, \mathbf{t}_i) \in \mathcal{P} \wedge \mathbf{t}_i \boldsymbol{\sigma}_i \to_{\mathcal{R}}^{*} \mathbf{s}_{i+1} \boldsymbol{\sigma}_{i+1}.$$

We use the abbreviation $\mathsf{ichain}\ (\mathcal{P},\, \mathcal{R})\ \mathbf{s}\ \mathbf{t}\ \boldsymbol{\sigma}$ for such a sequence. The soundness result of DPs then states that a (well-formed) TRS $\mathcal{R}$ is terminating if there is no infinite $(\mathsf{DP}(\mathcal{R}), \mathcal{R})$-chain where the formalization was described in [10].

Lemma 6. $\mathsf{wf\_trs}\ \mathcal{R} \Longrightarrow \neg\ \mathsf{SN}(\to_{\mathcal{R}}) \Longrightarrow \exists \mathbf{s}\ \mathbf{t}\ \boldsymbol{\sigma}.\ \mathsf{ichain}\ (\mathsf{DP}(\mathcal{R}),\, \mathcal{R})\ \mathbf{s}\ \mathbf{t}\ \boldsymbol{\sigma}$ □

Sometimes, we are interested in *minimal* $(\mathcal{P}, \mathcal{R})$-chains. The only difference between $\mathsf{min\_ichain}\ (\mathcal{P},\, \mathcal{R})\ \mathbf{s}\ \mathbf{t}\ \boldsymbol{\sigma}$ and $\mathsf{ichain}\ (\mathcal{P},\, \mathcal{R})\ \mathbf{s}\ \mathbf{t}\ \boldsymbol{\sigma}$, is the additional requirement in minimal chains that $\mathsf{SN}_{\mathcal{R}}(\mathbf{t}_i \boldsymbol{\sigma}_i)$ for all i.

4 Main Results

Since our term data type does not take care of building only terms corresponding to a specific signature, by default any rewrite relation $\to_{\mathcal{R}}$ in our formalization is defined over terms containing arbitrary function symbols. Our first goal is to show that once we have shown termination for terms using only function symbols from $\mathcal{F}(\mathcal{R})$, this implies that $\to_{\mathcal{R}}$ does terminate for arbitrary terms. Before doing that, we need means to identify well-formed terms. To this end we use the inductively defined set $\mathcal{T}(\mathcal{F})$, containing all terms that are well-formed with respect to the signature $\mathcal{F}$.

Definition 7 (Well-Formed Terms)

$$\frac{}{x \in \mathcal{T}(\mathcal{F})} \qquad \frac{(f,\, |\vec{ts}|) \in \mathcal{F} \quad \forall\, t \in \vec{ts}.\ t \in \mathcal{T}(\mathcal{F})}{f(\vec{ts}) \in \mathcal{T}(\mathcal{F})}$$

Using this definition we can define the well-formed rewrite relation induced by a TRS $\mathcal{R}$:

$$\Rrightarrow_{\mathcal{R}} \equiv \{(s,\, t) \mid s \to_{\mathcal{R}} t \wedge s \in \mathcal{T}(\mathcal{F}(\mathcal{R})) \wedge t \in \mathcal{T}(\mathcal{F}(\mathcal{R}))\}.$$

Further, let $\mathcal{C}(\mathcal{F})$ denote the set of well-formed contexts with respect to the signature $\mathcal{F}$. What we want to show is $\mathsf{SN}(\Rrightarrow_{\mathcal{R}}) \Longrightarrow \mathsf{SN}(\to_{\mathcal{R}})$. For the proof we need a way to remove unwanted function symbols from terms. This is the purpose of the following *cleaning* function:

$$\begin{aligned} [\![y]\!]_{\mathcal{F}} &= y \\ [\![f(\vec{ts})]\!]_{\mathcal{F}} &= \mathit{if}\ (f,\, |\vec{ts}|) \in \mathcal{F}\ \mathit{then}\ f(\mathit{map}\ [\![\cdot]\!]_{\mathcal{F}}\ \vec{ts})\ \mathit{else}\ z \end{aligned}$$

where z denotes an arbitrary but fixed variable. Intuitively, every subterm of a term whose root is not in the given signature, is replaced by z. Having this, the proof of $\mathsf{SN}(\Rrightarrow_{\mathcal{R}}) \Longrightarrow \mathsf{SN}(\rightarrow_{\mathcal{R}})$ (actually we prove its contrapositive) is done in three stages:

1. First, we assume $\neg\ \mathsf{SN}(\rightarrow_{\mathcal{R}})$. Then by the soundness of DPs (Lemma 6) we obtain an infinite $(\mathsf{DP}(\mathcal{R}),\ \mathcal{R})$-chain.
2. Next, we show that every infinite chain can be transformed into an infinite *clean* chain.
3. And finally, we show completeness of the DP-transformation for well-formed terms, i.e., that an infinite clean $(\mathsf{DP}(\mathcal{R}),\ \mathcal{R})$-chain can be transformed into an infinite derivation w.r.t. $\Rrightarrow_{\mathcal{R}}$. Hence, $\neg\ \mathsf{SN}(\Rrightarrow_{\mathcal{R}})$, concluding the proof.

In total we get $\mathsf{wf\_trs}\ \mathcal{R} \Longrightarrow \mathsf{SN}(\Rrightarrow_{\mathcal{R}}) \Longrightarrow \mathsf{SN}(\rightarrow_{\mathcal{R}})$ and since every non-well-formed TRS is nonterminating, we finally have a proof of Theorem 1. Note that the second step also shows the second main result: signature extensions are valid when performing termination proofs using DPs (without minimality).

It remains to prove the following two lemmas:

Lemma 8 (Signature Restrictions for Chains)

$$\mathcal{F}(\mathcal{P},\ \mathcal{R}) \subseteq \mathcal{F} \Longrightarrow \mathsf{ichain}\ (\mathcal{P},\ \mathcal{R})\ \mathbf{s}\ \mathbf{t}\ \boldsymbol{\sigma} \Longrightarrow \mathsf{ichain}\ (\mathcal{P},\ \mathcal{R})\ \mathbf{s}\ \mathbf{t}\ [\![\boldsymbol{\sigma}]\!]_{\mathcal{F}}$$

Lemma 9 (Completeness of DPs for $\Rrightarrow_{\mathcal{R}}$)

$$\mathsf{ichain}\ (\mathsf{DP}(\mathcal{R}),\ \mathcal{R})\ \mathbf{s}\ \mathbf{t}\ [\![\boldsymbol{\sigma}]\!]_{\sharp(\mathcal{R})} \Longrightarrow \neg\ \mathsf{SN}(\Rrightarrow_{\mathcal{R}})$$

where we use the abbreviations $\mathcal{F}(\mathcal{P},\ \mathcal{R}) \equiv \mathcal{F}(\mathcal{P}) \cup \mathcal{F}(\mathcal{R})$ and $\sharp(\mathcal{R}) \equiv \mathcal{F}(\mathcal{R}) \cup \mathcal{F}^{\sharp}(\mathcal{R})$ with $\mathcal{F}^{\sharp}(\mathcal{R}) \equiv \{(f^{\sharp},\ n) \mid (f,\ n) \in \mathcal{F}(\mathcal{R})\}$, and the cleaning function is extended to sequences of substitutions in the obvious way.

Note that by applying first Lemma 8 and then Lemma 9, we also obtain the classical completeness result of DPs.

Lemma 10 (Completeness of DPs). $\mathsf{ichain}\ (\mathsf{DP}(\mathcal{R}), \mathcal{R})\ \mathbf{s}\ \mathbf{t}\ \boldsymbol{\sigma} \Longrightarrow \neg\ \mathsf{SN}(\rightarrow_{\mathcal{R}})$

Proof. Obviously, we have $\mathcal{F}(\mathsf{DP}(\mathcal{R}),\ \mathcal{R}) \subseteq \sharp(\mathcal{R})$. Together with the assumption $\mathsf{ichain}\ (\mathsf{DP}(\mathcal{R}),\ \mathcal{R})\ \mathbf{s}\ \mathbf{t}\ \boldsymbol{\sigma}$, this yields $\mathsf{ichain}\ (\mathsf{DP}(\mathcal{R}),\ \mathcal{R})\ \mathbf{s}\ \mathbf{t}\ [\![\boldsymbol{\sigma}]\!]_{\sharp(\mathcal{R})}$, using Lemma 8. Then, from Lemma 9, we obtain $\neg\ \mathsf{SN}(\Rrightarrow_{\mathcal{R}})$ and thus $\neg\ \mathsf{SN}(\rightarrow_{\mathcal{R}})$. □

Proof (of Lemma 8). From the assumptions of Lemma 8 we obtain

$$\forall i.\, \mathbf{s}_i \in \mathcal{T}(\mathcal{F}) \wedge \mathbf{t}_i \in \mathcal{T}(\mathcal{F}), \tag{1}$$

$$\forall i.\, \mathbf{t}_i\boldsymbol{\sigma}_i \rightarrow^*_{\mathcal{R}} \mathbf{s}_{i+1}\boldsymbol{\sigma}_{i+1}, \tag{2}$$

$$\forall i.\, (\mathbf{s}_i, \mathbf{t}_i) \in \mathcal{P}. \tag{3}$$

Further note that whenever there is an $\mathcal{R}$-step from s to t, then either this step is also possible in the cleaned versions of s and t, or the cleaned versions are equal, i.e.,

$$s \rightarrow_{\mathcal{R}} t \Longrightarrow [\![s]\!]_{\mathcal{F}(\mathcal{R})} \rightarrow^{=}_{\mathcal{R}} [\![t]\!]_{\mathcal{F}(\mathcal{R})}.$$

From this and (2) we may conclude

$$\forall i.\, [\![\mathbf{t}_i \boldsymbol{\sigma}_i]\!]_{\mathcal{F}} \rightarrow^*_{\mathcal{R}} [\![\mathbf{s}_{i+1} \boldsymbol{\sigma}_{i+1}]\!]_{\mathcal{F}}$$

by induction over the length of the rewrite sequence (remember that $\mathcal{F}(\mathcal{R}) \subseteq \mathcal{F}$). Using (1) we may push the applications of the clean function inside, resulting in

$$\forall i.\, [\![\mathbf{t}_i]\!]_{\mathcal{F}} [\![\boldsymbol{\sigma}_i]\!]_{\mathcal{F}} \rightarrow^*_{\mathcal{R}} [\![\mathbf{s}_{i+1}]\!]_{\mathcal{F}} [\![\boldsymbol{\sigma}_{i+1}]\!]_{\mathcal{F}}.$$

Together with (3) we obtain the desired clean infinite chain as (1) shows $[\![\mathbf{s}_i]\!]_{\mathcal{F}} = \mathbf{s}_i$ and $[\![\mathbf{t}_i]\!]_{\mathcal{F}} = \mathbf{t}_i$ for all i. □

Proof (of Lemma 9). Again, we show the lemma in its contrapositive form. Thus, we assume $\mathsf{SN}(\Rrightarrow_{\mathcal{R}})$. Now, let $\mathcal{F}$ denote the signature of $\mathcal{R}$ and $\mathsf{u}(\cdot)$ the operation of 'unsharping,' i.e., removing ♯s from terms:

$$\mathsf{u}(t) = \begin{cases} f(\mathit{map}\ \mathsf{u}(\cdot)\ \vec{ts}) & \text{if } t = f^\sharp(\vec{ts}) \text{ or } t = f(\vec{ts}) \text{, and} \\ t & \text{otherwise.} \end{cases}$$

The extension of u to substitutions is defined as $\mathsf{u}(\sigma)(x) = \mathsf{u}(\sigma(x))$. For the sake of a contradiction, assume that there is an infinite $(\mathsf{DP}(\mathcal{R}), \mathcal{R})$-chain over $\mathbf{s}$, $\mathbf{t}$, and $[\![\boldsymbol{\sigma}]\!]_{\sharp(\mathcal{R})}$. Since cleaning does not affect $\mathbf{s}$ and $\mathbf{t}$, this implies an infinite $(\mathsf{DP}(\mathcal{R}), \mathcal{R})$-chain over $[\![\mathbf{s}]\!]_{\sharp(\mathcal{R})}$, $[\![\mathbf{t}]\!]_{\sharp(\mathcal{R})}$, and $[\![\boldsymbol{\sigma}]\!]_{\sharp(\mathcal{R})}$, i.e.,

$$\forall i.\, ([\![\mathbf{s}_i]\!]_{\sharp(\mathcal{R})}, [\![\mathbf{t}_i]\!]_{\sharp(\mathcal{R})}) \in \mathsf{DP}(\mathcal{R}) \tag{4}$$

$$\forall i.\, [\![\mathbf{t}_i]\!]_{\sharp(\mathcal{R})} [\![\boldsymbol{\sigma}_i]\!]_{\sharp(\mathcal{R})} \rightarrow^*_{\mathcal{R}} [\![\mathbf{s}_{i+1}]\!]_{\sharp(\mathcal{R})} [\![\boldsymbol{\sigma}_{i+1}]\!]_{\sharp(\mathcal{R})} \tag{5}$$

Then from (4) we obtain

$$\forall i.\, \exists C.\, C \in \mathcal{C}(\mathcal{F}) \wedge [\![\mathsf{u}(\mathbf{s}_i)]\!]_{\mathcal{F}} \rightarrow_{\mathcal{R}} C[[\![\mathsf{u}(\mathbf{t}_i)]\!]_{\mathcal{F}}]$$

by construction of $\mathsf{DP}(\mathcal{R})$. Using the *Axiom of Choice* we hence obtain a sequence of contexts $\mathbf{C}$, such that $\mathbf{C}_i$ is the context employed in the i-th step of (4), i.e.,

$$\forall i.\, \mathbf{C}_i \in \mathcal{C}(\mathcal{F}) \wedge [\![\mathsf{u}(\mathbf{s}_i)]\!]_{\mathcal{F}} \rightarrow_{\mathcal{R}} \mathbf{C}_i[[\![\mathsf{u}(\mathbf{t}_i)]\!]_{\mathcal{F}}] \tag{6}$$

Let $\mathbf{D}$ denote the following sequence:

$$\mathbf{D}_i = \begin{cases} \Box & \text{if } i = 0, \\ \mathbf{D}_{i-1} \circ (\mathbf{C}_i [\![\mathsf{u}(\boldsymbol{\sigma}_i)]\!]_{\mathcal{F}}) & \text{otherwise.} \end{cases}$$

Where $\circ$ denotes the composition of contexts, i.e., the right context replaces the hole of the left one. This function gives for the i-th DP-step in the infinite chain, all the contexts that have been lost due to using $\mathsf{DP}(\mathcal{R})$ instead of $\mathcal{R}$ and additionally applies all the necessary substitutions. For the sake of brevity we define:

$$\mathbf{s}'_i = \mathbf{D}_i[[\![\mathsf{u}(\mathbf{s}_i)]\!]_{\mathcal{F}} [\![\mathsf{u}(\boldsymbol{\sigma}_i)]\!]_{\mathcal{F}}]$$
$$\mathbf{t}'_i = \mathbf{D}_{i+1}[[\![\mathsf{u}(\mathbf{t}_i)]\!]_{\mathcal{F}} [\![\mathsf{u}(\boldsymbol{\sigma}_i)]\!]_{\mathcal{F}}]$$

Then by (6) we have $\mathbf{s}'_i \to_{\mathcal{R}} \mathbf{t}'_i$, since rewriting is closed under contexts and substitutions. From (5) we conclude $[\![\mathsf{u}(\mathbf{t}_i)]\!]_{\mathcal{F}}[\![\mathsf{u}(\boldsymbol{\sigma}_i)]\!]_{\mathcal{F}} \to^*_{\mathcal{R}} [\![\mathsf{u}(\mathbf{s}_{i+1})]\!]_{\mathcal{F}}[\![\mathsf{u}(\boldsymbol{\sigma}_{i+1})]\!]_{\mathcal{F}}$, since removing $\sharp$s does not destroy any redexes of $\mathcal{R}$. By wrapping this derivation in the context $\mathbf{D}_{i+1}$ we obtain $\mathbf{t}'_i \to^*_{\mathcal{R}} \mathbf{s}'_{i+1}$. Combining this with $\mathbf{s}'_i \to_{\mathcal{R}} \mathbf{t}'_i$ yields

$$\mathbf{s}'_i \to^+_{\mathcal{R}} \mathbf{s}'_{i+1}$$

From our assumption $\mathsf{SN}(\Rrightarrow_{\mathcal{R}})$ we conclude that $\mathcal{R}$ is well-formed. Moreover, it is apparent from the definitions of $[\![\cdot]\!]_{\mathcal{F}}$ and $\mathbf{D}_i$, together with (6) that all the $\mathbf{s}'_i$s are well-formed, i.e., $\mathbf{s}'_i \in \mathcal{T}(\mathcal{F})$. Together with the well-formedness of $\mathcal{R}$ one can prove that also all intermediate terms in all derivations $\mathbf{s}'_i \to^+_{\mathcal{R}} \mathbf{s}'_{i+1}$ are in $\mathcal{T}(\mathcal{F})$. Thus we have an infinite $\Rrightarrow_{\mathcal{R}}$-sequence which contradicts our assumption $\mathsf{SN}(\Rrightarrow_{\mathcal{R}})$. □

5 Applications

In most termination tools, termination techniques are freely combined within a complex termination proof. For example, it is a standard procedure to first remove some rules from $\mathcal{R}$, resulting in $\mathcal{R}'$, and then prove $\mathsf{SN}(\mathcal{R}')$ without caring about any changes in the signature. I.e., proving termination of $\mathsf{SN}(\mathcal{R}')$ is performed as if the signature were $\mathcal{F}(\mathcal{R}')$ and not the original signature $\mathcal{F}(\mathcal{R})$. The soundness of this approach relies upon Theorem 1.

At first view, Theorem 1 might not seem important, as there are several termination techniques which do not rely upon the signature. For example, when using polynomial interpretations, it always suffices to give the interpretations for the function symbols occurring in the TRS, no matter if the signature contains other symbols. The reason is that the interpretation of any other symbol has no impact when computing the polynomials for the left-hand sides and right-hand sides of the rules. Similar situations occur for other reduction orders and other termination techniques, like semantic labeling [11].

However, we are aware of at least two termination techniques where the signature is essential.

String Reversal. If we restrict terms in rewriting to employ only unary function symbols, we are in the setting of *string rewriting*. For notational convenience we write abc instead of $\mathsf{a}(\mathsf{b}(\mathsf{c}(x)))$, where the variable x is implicit. There are several termination techniques that work only/better for strings. One of them is string reversal. This technique uses the fact that a string rewrite system (SRS) $\mathcal{S}$ is terminating iff $\mathsf{rev}(\mathcal{S})$ is terminating. Here, $\mathsf{rev}(\mathcal{S})$ denotes the mapping of the function

$$\mathsf{rev}(t) = \begin{cases} \mathsf{rev}(t')a & \text{if } t = at', \\ t & \text{otherwise,} \end{cases}$$

over all left-hand sides and right-hand sides of $\mathcal{S}$. In practice, this often helps to automatically find a termination proof.

Example 11. Consider the following TRS

$$\mathsf{a}(\mathsf{b}(\mathsf{b}(x))) \to \mathsf{a}(\mathsf{b}(\mathsf{a}(\mathsf{a}(\mathsf{a}(\mathsf{a}(x))))))$$
$$\mathsf{f}(x, y) \to x$$

which is not an SRS. One can remove the second rule by a polynomial order which maps $\mathsf{a}(x)$ and $\mathsf{b}(x)$ to x, and $\mathsf{f}(x, y)$ to $x + y + 1$. Then the SRS

$$\mathsf{abb} \to \mathsf{abaaaa}$$

remains, where the signature still contains the binary symbol f. As string reversal is only defined for unary symbols, the presence of f forbids the application of string reversal. But after applying the signature restriction to a and b we are allowed to forget about f and apply string reversal to obtain the following SRS:

$$\mathsf{bba} \to \mathsf{aaaaba}$$

Note that in this reversed SRS there are no dependency pairs as ba is a proper subterm of bba. Therefore, termination is now trivially proven.

Root-Labeling. Root-labeling [8] is a special version of semantic labeling [11]. We start with a short description of semantic labeling. We interpret a TRS $\mathcal{R}$ by an $\mathcal{F}$-algebra $\mathcal{M} = (M, \{f_\mathcal{M}\}_{f \in \mathcal{F}})$. That is, we interpret every function symbol f of arity n, by a function $f_\mathcal{M} \colon M^n \to M$, over the carrier M. Then, the interpretation of a term with respect to a given variable assignment μ, is given by: $[\mu]_\mathcal{M}(x) = \mu(x)$ and $[\mu]_\mathcal{M}(f(t_1, \ldots, t_n)) = f_\mathcal{M}([\mu]_\mathcal{M}(t_1), \ldots, [\mu]_\mathcal{M}(t_n))$. We say that $\mathcal{M}$ is a *model* of $\mathcal{R}$ iff for all assignments μ and all rules $l \to r \in \mathcal{R}$, we have $[\mu]_\mathcal{M}(l) = [\mu]_\mathcal{M}(r)$.

Now, we can label the function symbols of $\mathcal{R}$ according to the interpretation of their arguments. For every n-ary function symbol f, we choose a set of labels $L_f \neq \varnothing$ in combination with a mapping $\pi_f \colon M^n \to L_f$. The labeling is extended to terms as follows: $\mathsf{lab}_\mu(x) = x$ and $\mathsf{lab}_\mu(f(t_1, \ldots, t_n)) = f_m(\mathsf{lab}_\mu(t_1), \ldots, \mathsf{lab}_\mu(t_n))$ with $m = \pi_f([\mu]_\mathcal{M}(t_1), \ldots, [\mu]_\mathcal{M}(t_n))$. Then, the labeled TRS $\mathcal{R}_\mathsf{lab}$ consists of the rules $\mathsf{lab}_\mu(l) \to \mathsf{lab}_\mu(r)$ for all $l \to r \in \mathcal{R}$ and assignments μ. Zantema [11] has shown that for every model of $\mathcal{R}$, the TRS $\mathcal{R}$ is terminating iff the TRS $\mathcal{R}_\mathsf{lab}$ is terminating.

The difficult part of applying semantic labeling for proving termination, is to find a proper model. This is solved in the special case of root-labeling by fixing the interpretation. Every function symbol is interpreted by itself (i.e., $f_\mathcal{M}(x_1, \ldots, x_n) = f$) and the labeling is fixed to tuples of function symbols (i.e., $\pi_f(x_1, \ldots, x_n) = (x_1, \ldots, x_n)$). Now, to satisfy the necessary *model condition*, we close the rules of a TRS under the so called *flat contexts* before labeling. This makes sure that for every resulting rule $l \to r$, the root symbol of l is the same as the root symbol of r and thus, $[\mu]_\mathcal{M}(l) = [\mu]_\mathcal{M}(r)$ for every assignment μ. Here, the flat contexts are determined solely by the signature. Again, Theorem 1 shows that one can reduce the possibly infinite set of flat contexts (if the signature $\mathcal{F}$ is infinite) to a finite set of flat contexts (if $\mathcal{F}(\mathcal{R})$ is finite).

Example 12. Consider the TRS $\{\mathsf{a}(\mathsf{b}(x)) \to \mathsf{b}(\mathsf{a}(\mathsf{a}(x)))\}$. This yields the set of flat contexts $\{\mathsf{a}(\Box), \mathsf{b}(\Box)\}$. After closing the TRS under flat contexts we obtain the two rules $\{\mathsf{a}(\mathsf{a}(\mathsf{b}(x))) \to \mathsf{a}(\mathsf{b}(\mathsf{a}(\mathsf{a}(x)))), \mathsf{b}(\mathsf{a}(\mathsf{b}(x))) \to \mathsf{b}(\mathsf{b}(\mathsf{a}(\mathsf{a}(x))))\}$.[3] Now, root-labeling results in the following labeled TRS:

$$\begin{aligned}
\mathsf{a}_\mathsf{a}(\mathsf{a}_\mathsf{b}(\mathsf{b}_\mathsf{a}(x))) &\to \mathsf{a}_\mathsf{b}(\mathsf{b}_\mathsf{a}(\mathsf{a}_\mathsf{a}(\mathsf{a}_\mathsf{a}(x))))\\
\mathsf{a}_\mathsf{a}(\mathsf{a}_\mathsf{b}(\mathsf{b}_\mathsf{b}(x))) &\to \mathsf{a}_\mathsf{b}(\mathsf{b}_\mathsf{a}(\mathsf{a}_\mathsf{a}(\mathsf{a}_\mathsf{b}(x))))\\
\mathsf{b}_\mathsf{a}(\mathsf{a}_\mathsf{b}(\mathsf{b}_\mathsf{a}(x))) &\to \mathsf{b}_\mathsf{b}(\mathsf{b}_\mathsf{a}(\mathsf{a}_\mathsf{a}(\mathsf{a}_\mathsf{a}(x))))\\
\mathsf{b}_\mathsf{a}(\mathsf{a}_\mathsf{b}(\mathsf{b}_\mathsf{b}(x))) &\to \mathsf{b}_\mathsf{b}(\mathsf{b}_\mathsf{a}(\mathsf{a}_\mathsf{a}(\mathsf{a}_\mathsf{b}(x))))
\end{aligned}$$

The advantage of root-labeling or semantic labeling is that afterwards one can distinguish different occurrences of symbols as they might have different labels. For example, the last but one symbols of the left- and right-hand sides are a_b and a_a, respectively, whereas in the original TRS these symbols were just a and could not be distinguished. That such a distinction can be helpful is demonstrated in several examples [8,11].

Note that root-labeling is also applied in the DP setting. Here, Lemma 8 can be used to show that it suffices to build the flat contexts w.r.t. the signature of the given DP problem.

However, many termination tools base their termination analysis on DPs where always minimal chains are considered. The reason to work with minimal chains is that many powerful termination techniques are only sound when regarding minimal chains [4,9].

Unfortunately, when trying to lift Lemma 8 to minimal chains, we figured out that this is impossible. It is easy to show that cleaning terms might introduce nontermination if non-left-linear rules are present. For example if a and b are not in the signature and there is a rule $\mathsf{f}(x, x) \to \mathsf{f}(x, x)$, then this rule cannot be applied on $\mathsf{f}(\mathsf{a}, \mathsf{b})$. However, it is applicable on the cleaned term $\mathsf{f}(z, z)$.

Moreover, we even found a counter-example where there is an infinite minimal $(\mathcal{P}, \mathcal{R})$-chain, but no infinite minimal $(\mathcal{P}, \mathcal{R})$-chain if only terms over $\mathcal{F}(\mathcal{P} \cup \mathcal{R})$ may be used. Hence, there cannot be any function that transforms an infinite minimal $(\mathcal{P}, \mathcal{R})$-chain over an arbitrary signature into an infinite minimal chain which only contains terms over $\mathcal{F}(\mathcal{P} \cup \mathcal{R})$. In other words, Lemma 8 does not hold if one would replace infinite chains by minimal infinite chains.

Example 13 (Restricting the signature to $\mathcal{F}(\mathcal{P} \cup \mathcal{R})$ is unsound for minimal chains). To present a counter-example we give a "termination proof" for a non-terminating TRS where the only unsound step is the signature restriction to the signature of the current DP problem. Here, we make use of the DP-framework [4] in which one proves termination by simplifying the initial DP problems by termination techniques until one obtains a DP problem that does not admit an

[3] If the signature would be larger, e.g., if there would be an additional ternary symbol c, then the flat contexts would include $\{\mathsf{c}(\Box, x_2, x_3), \mathsf{c}(x_1, \Box, x_3), \mathsf{c}(x_1, x_2, \Box)\}$ and for each of these contexts one would get another rule. Hence, the signature restriction is essential to get few flat contexts and therefore small systems.

infinite minimal chain. For soundness it is only required that whenever $(\mathcal{P}, \mathcal{R})$ is simplified to $(\mathcal{P}', \mathcal{R}')$ then an infinite minimal $(\mathcal{P}, \mathcal{R})$-chain must imply the existence of an infinite minimal $(\mathcal{P}', \mathcal{R}')$-chain.

So, let $\mathcal{R}$ be the following nonterminating TRS.

$$
\begin{aligned}
\mathsf{g}(\mathsf{f}(x, y, x', z, z, u)) &\to \mathsf{g}(\mathsf{f}(x, y, x, x, y, \mathsf{h}(y, x'))) \\
\mathsf{a} &\to \mathsf{b} \\
\mathsf{a} &\to \mathsf{c} \\
\mathsf{h}(x, x) &\to \mathsf{h}(x, x) \\
\mathsf{h}(\mathsf{a}, x) &\to \mathsf{h}(x, x) \\
\mathsf{h}(\mathsf{b}, x) &\to \mathsf{h}(x, x) \\
\mathsf{h}(\mathsf{c}, x) &\to \mathsf{h}(x, x) \\
\mathsf{h}(\mathsf{h}(x_1, x_2), x) &\to \mathsf{h}(x, x) \\
\mathsf{h}(\mathsf{f}(x_1, \dots, x_6), x) &\to \mathsf{h}(x, x)
\end{aligned}
$$

The initial DP problem $(\mathsf{DP}(\mathcal{R}), \mathcal{R})$ can be simplified to $(\mathcal{P}, \mathcal{R})$ where $\mathcal{P} = \{\mathsf{G}(\mathsf{f}(x, y, x', z, z, u)) \to \mathsf{G}(\mathsf{f}(x, y, x, x, y, \mathsf{h}(y, x')))\}$. The reason is that there is a minimal infinite $(\mathcal{P}, \mathcal{R})$-chain: choose every $\mathbf{s}_i$ and $\mathbf{t}_i$ to be the left-hand side and right-hand side of the only rule in $\mathcal{P}$, respectively. Further, choose $\boldsymbol{\sigma}_i = \sigma$ for $\sigma(x) = \mathsf{g}(\mathsf{a})$, $\sigma(y) = \sigma(z) = \mathsf{g}(\mathsf{b})$, $\sigma(x') = \mathsf{g}(\mathsf{c})$, and $\sigma(u) = \mathsf{h}(\mathsf{g}(\mathsf{b}), \mathsf{g}(\mathsf{c}))$.

Note that this chain is also a minimal $(\mathcal{P}, \mathcal{R}')$-chain where $\mathcal{R}'$ is like $\mathcal{R}$ but without the $\mathsf{g}(\dots) \to \mathsf{g}(\dots)$-rule. Thus, $(\mathcal{P}, \mathcal{R})$ can be simplified to $(\mathcal{P}, \mathcal{R}')$.

Using the argument filter processor [9, Theorem 4.37], it is shown that there also is an infinite minimal chain when collapsing G to its first argument. Hence, the same substitution σ can be used to obtain an infinite minimal chain for the DP problem $(\mathcal{P}', \mathcal{R}')$ where $\mathcal{P}' = \{\mathsf{f}(x, y, x', z, z, u) \to \mathsf{f}(x, y, x, x, y, \mathsf{h}(y, x'))\}$.

Now, if it would be sound to restrict the signature of $(\mathcal{P}', \mathcal{R}')$ to $\mathcal{F}(\mathcal{P}' \cup \mathcal{R}') = \{\mathsf{a}, \mathsf{b}, \mathsf{c}, \mathsf{f}, \mathsf{h}\}$ then we can conclude termination. The reason is that over this signature there are no infinite minimal $(\mathcal{P}', \mathcal{R}')$-chains anymore.

We prove this last statement by contradiction. Suppose there is an infinite minimal $(\mathcal{P}', \mathcal{R}')$-chain over $\mathbf{s}$, $\mathbf{t}$, and $\boldsymbol{\delta}$, where $\boldsymbol{\delta}_i$ instantiates all variables by terms over $\mathcal{F}(\mathcal{P}' \cup \mathcal{R}')$, $\mathbf{s}_i = \mathsf{f}(x, y, x', z, z, u)$, and $\mathbf{t}_i = \mathsf{f}(x, y, x, x, y, \mathsf{h}(y, x'))$, for all i. $\boldsymbol{\delta}_i$. Then all $\mathbf{t}_i\boldsymbol{\delta}_i$ are terminating w.r.t. $\mathcal{R}'$. Hence, $\boldsymbol{\delta}_1(y)$ must be a variable (otherwise, $\mathsf{h}(y, x')\boldsymbol{\delta}_1$ would not be terminating due to the six h-rules of $\mathcal{R}'$). Moreover, by using that $\boldsymbol{\delta}_1(y)$ is a variable, the derivation $\mathbf{t}_1\boldsymbol{\delta}_1 \to_{\mathcal{R}'}^* \mathbf{s}_2\boldsymbol{\delta}_2$ shows that $\boldsymbol{\delta}_1(y) = \boldsymbol{\delta}_2(y)$ and $\boldsymbol{\delta}_1(y) = \boldsymbol{\delta}_2(z)$. Note that since $\mathcal{R}'$ is not collapsing, whenever a term rewrites to a variable then the term must be identical to the variable. Hence, since $\boldsymbol{\delta}_2(z)$ is a variable and $\boldsymbol{\delta}_1(x) \to_{\mathcal{R}'}^* \boldsymbol{\delta}_2(z)$ we obtain $\boldsymbol{\delta}_1(x) = \boldsymbol{\delta}_2(z)$ and for a similar reason we obtain $\boldsymbol{\delta}_1(x) = \boldsymbol{\delta}_2(x')$. In total, we can conclude $\boldsymbol{\delta}_2(y) = \boldsymbol{\delta}_2(x')$. This finally yields a contradiction as there is the nonterminating subterm $\mathsf{h}(y, x')\boldsymbol{\delta}_2 = \mathsf{h}(\boldsymbol{\delta}_2(y), \boldsymbol{\delta}_2(y))$.

The consequences are severe: termination proofs relying upon techniques that require minimal chains and also use signature restrictions are unsound without further restrictions.

And indeed, for the technique of root-labeling—which performs a signature restriction within the soundness proof—we were able to construct a counter-example which refutes the main theorem for root-labeling with DPs.

Example 14 (Root-labeling is unsound for minimal chains). We use a similar TRS as in Example 13 to show the problem of root-labeling with minimal chains. Let $\mathcal{R}$ consist of the following rules.

$$\begin{aligned} \mathsf{g}(\mathsf{f}(x, y, x', z, z, u)) &\to \mathsf{g}(\mathsf{f}(x, y, x, x, y, \mathsf{h}(y, x'))) \\ \mathsf{a} &\to \mathsf{b} \\ \mathsf{a} &\to \mathsf{c} \\ \mathsf{h}(x, x) &\to \mathsf{h}(x, x) \\ \mathsf{h}(\mathsf{a}, x) &\to \mathsf{h}(x, x) \\ \mathsf{h}(x, \mathsf{a}) &\to \mathsf{h}(x, x) \\ \mathsf{f}(x_1, \ldots, \mathsf{a}, \ldots, x_5) &\to \mathsf{f}(x_1, \ldots, \mathsf{a}, \ldots, x_5) \end{aligned}$$

Here, the last rule represents 6 rules where the a can be at any position.

We again can simplify the initial DP-problem $(\mathsf{DP}(\mathcal{R}), \mathcal{R})$ to $(\mathcal{P}, \mathcal{R})$ for the same $\mathcal{P} = \{\mathsf{G}(\mathsf{f}(x, y, x', z, z, u)) \to \mathsf{G}(\mathsf{f}(x, y, x, x, y, \mathsf{h}(y, x')))\}$ that we had in the previous example. The reason is again that there is an infinite minimal chain by choosing $\boldsymbol{\sigma}_i = \sigma$ for $\sigma(x) = \mathsf{g}(\mathsf{a})$, $\sigma(y) = \sigma(z) = \mathsf{g}(\mathsf{b})$, $\sigma(x') = \mathsf{g}(\mathsf{c})$, and $\sigma(u) = \mathsf{h}(\mathsf{g}(\mathsf{b}), \mathsf{g}(\mathsf{c}))$.

Note that by using this substitution we also get an infinite minimal $(\mathcal{P}, \mathcal{R}')$-chain where $\mathcal{R}' = \mathcal{R} \setminus \{\mathsf{g}(\dots) \to \mathsf{g}(\dots)\}$. Hence, it is sound to simplify $(\mathcal{P}, \mathcal{R})$ to $(\mathcal{P}, \mathcal{R}')$.

Now, in [8, proofs of Lemmas 13 and 17] it is wrongly stated[4] that for this example, *w.l.o.g. one can restrict to substitutions over the signature* $\{\mathsf{a}, \mathsf{b}, \mathsf{c}, \mathsf{f}, \mathsf{h}\}$: With a similar reasoning as in Example 13 one can prove that there is no infinite minimal $(\mathcal{P}, \mathcal{R}')$-chain using this restricted class of substitutions. We further show in detail that Lemma 17 of [8] itself is wrong, not only its proof.

The result of Lemma 17 states that if there is an infinite minimal $(\mathcal{P}, \mathcal{R}')$-chain then there also is an infinite minimal chain for the DP problem $(\mathcal{P}', \mathcal{R}'')$ that is obtained by the flat context closure. In our example we obtain $\mathcal{P}' = \mathcal{P} \cup \{\mathsf{G}(\mathsf{a}) \to \mathsf{G}(\mathsf{b}), \mathsf{G}(\mathsf{a}) \to \mathsf{G}(\mathsf{c})\}$ and $\mathcal{R}'' = (\mathcal{R}' \setminus \{\mathsf{a} \to \mathsf{b}, \mathsf{a} \to \mathsf{c}\}) \cup \mathcal{R}'''$ where $\mathcal{R}'''$ consists of the following rules:

$$\begin{aligned} \mathsf{h}(\mathsf{a}, x) &\to \mathsf{h}(\mathsf{b}, x) \\ \mathsf{h}(\mathsf{a}, x) &\to \mathsf{h}(\mathsf{c}, x) \\ \mathsf{h}(x, \mathsf{a}) &\to \mathsf{h}(x, \mathsf{b}) \\ \mathsf{h}(x, \mathsf{a}) &\to \mathsf{h}(x, \mathsf{c}) \\ \mathsf{f}(x_1, \ldots, \mathsf{a}, \ldots, x_5) &\to \mathsf{f}(x_1, \ldots, \mathsf{b}, \ldots, x_5) \\ \mathsf{f}(x_1, \ldots, \mathsf{a}, \ldots, x_5) &\to \mathsf{f}(x_1, \ldots, \mathsf{c}, \ldots, x_5) \end{aligned}$$

[4] In detail, in [8] our upcoming Lemma 17 is used without the requirement of left-linearity.

We show that for this DP problem $(\mathcal{P}', \mathcal{R}'')$ there are no infinite minimal chains anymore. So, if Lemma 17 of [8] would be sound, we could wrongly "prove" termination of $\mathcal{R}$. Again, we assume there is an infinite minimal $(\mathcal{P}', \mathcal{R}'')$-chain where $\boldsymbol{\delta}_i$ are the corresponding substitutions and where we do not even restrict the signature of any $\boldsymbol{\delta}_i$. Obviously, all $(\mathbf{s}_i, \mathbf{t}_i)$ are taken from $\mathcal{P}$ and not from one of the additional rules in $\mathcal{P}'$. Since every left-hand side of $\mathcal{R}''$ also is a left-hand side of a nonterminating rule in $\mathcal{R}''$, we know that every terminating term w.r.t. $\mathcal{R}''$ is also a normal form w.r.t. $\mathcal{R}''$. Hence, from $\mathbf{t}_1\boldsymbol{\delta}_1 \to^*_{\mathcal{R}''} \mathbf{s}_2\boldsymbol{\delta}_2$ we conclude $\mathbf{t}_1\boldsymbol{\delta}_1 = \mathbf{s}_2\boldsymbol{\delta}_2$. Thus, $\boldsymbol{\delta}_2(x') = \boldsymbol{\delta}_1(x) = \boldsymbol{\delta}_2(z) = \boldsymbol{\delta}_1(y) = \boldsymbol{\delta}_2(y)$. Therefore, we obtain the nonterminating subterm $\mathsf{h}(y, x')\boldsymbol{\delta}_2 = \mathsf{h}(\boldsymbol{\delta}_2(y), \boldsymbol{\delta}_2(y))$ which is a contradiction to the minimality of the chain.

To conclude, the current applications of root-labeling in termination tools which rely upon DPs with minimal chains are wrong for two reasons: first, one cannot restrict the signature to the implicit signature of the given DP-problem, and second, root-labeling is unsound in the DP setting with minimal chains.

However, for signature restrictions in combination with minimal chains we were able to prove soundness, provided that the TRS $\mathcal{R}$ of a DP problem $(\mathcal{P}, \mathcal{R})$ is left-linear.

Lemma 15 (Signature Restrictions for Minimal Chains)
$\mathsf{left\_linear}\ \mathcal{R} \Longrightarrow$
$\mathcal{F}(\mathcal{P}, \mathcal{R}) \subseteq \mathcal{F} \Longrightarrow \mathsf{min\_ichain}\ (\mathcal{P}, \mathcal{R})\ \mathbf{s}\ \mathbf{t}\ \boldsymbol{\sigma} \Longrightarrow \mathsf{min\_ichain}\ (\mathcal{P}, \mathcal{R})\ \mathbf{s}\ \mathbf{t}\ [\![\boldsymbol{\sigma}]\!]_{\mathcal{F}}$

The proof of Lemma 15 is similar to the proof of Lemma 8. The only missing step is to prove that left-linearity ensures that cleaning does not introduce nontermination.

Lemma 16 (Cleaning of Left-Linear TRSs Preservers SN)

1. $\mathsf{left\_linear}\ \mathcal{R} \Longrightarrow \mathcal{F}(\mathcal{R}) \subseteq \mathcal{F} \Longrightarrow \mathsf{SN}_{\mathcal{R}}(s) \Longrightarrow [\![s]\!]_{\mathcal{F}} \to_{\mathcal{R}} t \Longrightarrow \exists\, u.\ [\![u]\!]_{\mathcal{F}} = t \wedge s \to_{\mathcal{R}} u$
2. $\mathsf{left\_linear}\ \mathcal{R} \Longrightarrow \mathcal{F}(\mathcal{R}) \subseteq \mathcal{F} \Longrightarrow \mathsf{SN}_{\mathcal{R}}(s) \Longrightarrow \mathsf{SN}_{\mathcal{R}}([\![s]\!]_{\mathcal{F}})$

Proof. 1. We prove this fact via induction over s. In the base case, s is a variable. Then we have the rewrite step $s \to_{\mathcal{R}} t$, since cleaning does not change variables. But then, there is a variable left-hand side, implying that $\mathcal{R}$ is not terminating and thus contradicting $\mathsf{SN}_{\mathcal{R}}(s)$.

In the step case we have $s = f(\vec{ss})$. Now, we proceed by a case distinction. If $(f, |\vec{ss}|) \notin \mathcal{F}$ then cleaning will transform s into the variable z. Again, there would be a variable left-hand side, contradicting strong normalization of s. Thus, $(f, |\vec{ss}|) \in \mathcal{F}$. Hence, $f(\mathit{map}\ [\![\cdot]\!]_{\mathcal{F}}\ \vec{ss}) \to_{\mathcal{R}} t$. If this is a non-root step, the result follows from the induction hypothesis. Otherwise, this is a root rewrite step. Thus we obtain a rule $(l, r) \in \mathcal{R}$ and a substitution σ, such that, $[\![f(\vec{ss})]\!]_{\mathcal{F}} = l\sigma$ and $r\sigma = t$. Additionally, we know that this rule is left-linear and that its left-hand side is well-formed. It can be shown that this implies the existence of a substitution τ, such that, $[\![\tau_{\mathcal{V}\mathsf{ar}(l)}]\!]_{\mathcal{F}} = \sigma|_{\mathcal{V}\mathsf{ar}(l)}$ and $f(\vec{ss}) = l\tau$ (we omit the rather technical proof). Here, $\sigma|_V$ denotes the restriction of a substitution σ to a set of variables V, i.e., all variables that are not in V, are no longer modified by the restricted substitution. Then $[\![r\tau]\!]_{\mathcal{F}} = [\![r]\!]_{\mathcal{F}}[\![\tau]\!]_{\mathcal{F}} = r[\![\tau_{\mathcal{V}\mathsf{ar}(l)}]\!]_{\mathcal{F}} = r\sigma|_{\mathcal{V}\mathsf{ar}(l)} = r\sigma = t$ and $s = f(\vec{ss}) = l\tau \to_{\mathcal{R}} r\tau$. Here, we needed to use the property $\mathcal{V}\mathsf{ar}(r) \subseteq \mathcal{V}\mathsf{ar}(l)$, which must be valid since otherwise $\mathsf{SN}_{\mathcal{R}}(s)$ does not hold.

2. Assume that $[\![s]\!]_{\mathcal{F}}$ is not terminating. Thus, there is an infinite sequence of $\mathcal{R}$-steps, starting from $[\![s]\!]_{\mathcal{F}}$. By iteratively applying the previous result, we obtain an infinite $\mathcal{R}$-sequence starting at s. □

We were also able to formally show that the signature restriction that is done in root-labeling (which is exactly the upcoming Lemma 17 without the requirement of left-linearity) is sound for minimal chains with the requirement of left-linearity. Hence, with the following lemma one can repair the paper proofs of [8, Lemmas 13 and 17] by demanding left-linearity. Essentially, the lemma states that one can restrict to the symbols that occur below the root in $\mathcal{P}$ ($\mathcal{F}_{>\epsilon}(\mathcal{P})$), together with the symbols of $\mathcal{R}$, under the additional assumption that neither left-hand sides nor right-hand sides of $\mathcal{P}$ are variables and the roots of $\mathcal{P}$ are not defined in $\mathcal{R}$.

Lemma 17 (Signature Restrictions Ignoring Roots)
$\mathsf{left\_linear}\ \mathcal{R} \Longrightarrow$
$\quad \mathcal{F}_{>\epsilon}(\mathcal{P}) \cup \mathcal{F}(\mathcal{R}) \subseteq \mathcal{F} \Longrightarrow$
$\quad \forall\, s\ t.\ (s,\, t) \in \mathcal{P} \longrightarrow s \notin \mathcal{V}\mathsf{ar} \wedge t \notin \mathcal{V}\mathsf{ar} \wedge \neg\ \mathsf{root}(t) \in \mathcal{D}(\mathcal{R}) \Longrightarrow$
$\quad \mathsf{min\_ichain}\ (\mathcal{P},\, \mathcal{R})\ \mathbf{s}\ \mathbf{t}\ \boldsymbol{\sigma} \Longrightarrow \mathsf{min\_ichain}\ (\mathcal{P},\, \mathcal{R})\ \mathbf{s}\ \mathbf{t}\ [\![\boldsymbol{\sigma}]\!]_{\mathcal{F}}$

The lemma is proven in the same way as Lemma 15, except that one only applies cleaning strictly below the root. By cleaning below the root one can also proof a variant of Lemma 17 where minimal chains are replaced by arbitrary chains, and where left-linearity is no longer required.[5]

Using Lemma 17 and the original proofs of [8] it is shown that root-labeling is sound in combination with minimal chains if we restrict to left-linear $\mathcal{R}$-components. Hence, the main example of [8, Touzet's SRS] is still working, since it applies root-labeling on a DP problem with left-linear $\mathcal{R}$.

6 Conclusion

We presented an alternative, and more importantly, the first mechanized proof of the fact that termination is preserved under signature extensions. We have also shown that signature extensions are possible when using DPs, but only if one considers arbitrary chains or left-linear TRSs. For minimal chains we have given a counterexample which shows that for non-left-linear TRSs one cannot restrict to the signature of the current DP problem.

We believe these results to be interesting in their own. However, we developed these results with a certain goal in mind. In the end we want to apply our

[5] However, one needs the additional requirement that left-hand sides of $\mathcal{R}$ are not variables, which in Lemma 17 follows from the minimality of the chain.

main positive results to be able to certify termination proofs which rely upon techniques where the signature is essential: string reversal and root-labeling. If one applies these techniques directly on a TRS, then both techniques can now be certified in the way they are used in current termination tools. For root-labeling in the DP setting with minimal chains, we have shown that it is unsound for arbitrary DP problems. We have further shown how to repair the existing proofs by demanding left-linearity. It remains as future work, to also formalize the remaining proof for root-labeling in the DP setting.

References

1. Arts, T., Giesl, J.: Termination of term rewriting using dependency pairs. Theoretical Computer Science 236, 133–178 (2000)
2. Baader, F., Nipkow, T.: Term Rewriting and All That. Cambridge University Press, Cambridge (1998)
3. Dershowitz, N.: Termination dependencies. In: Proc. WST 2003, pp. 27–30 (2003)
4. Giesl, J., Thiemann, R., Schneider-Kamp, P., Falke, S.: Mechanizing and improving dependency pairs. Journal of Automated Reasoning 37(3), 155–203 (2006)
5. Middeldorp, A.: Modular Properties of Term Rewriting Systems. PhD thesis, Vrije Universiteit, Amsterdam (1990)
6. Nipkow, T., Paulson, L., Wenzel, M.: Isabelle/HOL. LNCS, vol. 2283. Springer, Heidelberg (2002)
7. Ohlebusch, E.: A simple proof of sufficient conditions for the termination of the disjoint union of term rewriting systems. Bulletin of the EATCS 50, 223–228 (1993)
8. Sternagel, C., Middeldorp, A.: Root-Labeling. In: Voronkov, A. (ed.) RTA 2008. LNCS, vol. 5117, pp. 336–350. Springer, Heidelberg (2008)
9. Thiemann, R.: The DP Framework for Proving Termination of Term Rewriting. PhD thesis, RWTH Aachen University (2007), Available as Technical Report AIB-2007-17, `http://aib.informatik.rwth-aachen.de/2007/2007-17.pdf`
10. Thiemann, R., Sternagel, C.: Certification of termination proofs using CeTA. In: Berghofer, S., Nipkow, T., Urban, C., Wenzel, M. (eds.) TPHOLs 2009. LNCS, vol. 5674, pp. 452–468. Springer, Heidelberg (2009)
11. Zantema, H.: Termination of term rewriting by semantic labelling. Fundamenta Informaticae 24(1/2), 89–105 (1995)