Simulating Dependency Pairs by Semantic Labeling

Teppei Saito
JAIST, Nomi, Japan

Nao Hirokawa
JAIST, Nomi, Japan

Abstract

We show that termination proofs by a version of the dependency pair method can be simulated by semantic labeling plus multiset path orders. By incorporating a flattening technique into multiset path orders the simulation result can be extended to the dependency pair method for relative termination, introduced by Iborra et al. This result allows us to improve applicability of their dependency pair method.

2012 ACM Subject Classification Theory of computation → Equational logic and rewriting

Keywords and phrases Term rewriting, Relative termination, Semantic labeling, Dependency pairs

Digital Object Identifier 10.4230/LIPIcs.FSCD.2024.13

Supplementary Material *Software (Tool and Experimental Data)*:
`https://www.jaist.ac.jp/project/saigawa/24fscd/`

Funding *Teppei Saito*: JST SPRING Grant Number JPMJSP2102
Nao Hirokawa: JSPS KAKENHI Grant Number JP22K11900

Acknowledgements The second author is grateful to Vincent van Oostrom for his valuable questions and comments on his talk at FSCD 2021. We thank René Thiemann and Jürgen Giesl for telling us about the first correctness proof of the dependency pair method via semantic labeling. Last but certainly not least, Alfons Geser and Aart Middeldorp's help on literature was indispensable for this work.

1 Introduction

Arts and Giesl's *dependency pair method* [4] and Zantema's *semantic labeling* [29] are powerful techniques for analyzing termination of term rewrite systems (TRSs). In this paper we show that the former can be simulated by the latter combined with a restricted version of *multiset path orders* [8, Definition 5] (also known as recursive path orders).

Let us give an informal outlook of the idea by means of examples. The first example is a termination proof by the dependency pair method. *Dependency pairs* are rewrite rules that represent dependencies of recursive function calls in a TRS. Termination of the TRS boils down to the problem of finding a suitable well-founded algebra with interpretations that weakly orient all rules in the TRS and strictly orient all dependency pairs.

▶ **Example 1.** We show the termination of the TRS for division of Peano numbers:

$$x - \mathsf{0} \to x \qquad \mathsf{s}(x) - \mathsf{s}(y) \to x - y \qquad \mathsf{0} \div \mathsf{s}(y) \to \mathsf{0} \qquad \mathsf{s}(x) \div \mathsf{s}(y) \to \mathsf{s}((x - y) \div \mathsf{s}(y))$$

There are three dependency pairs:

$$\mathsf{s}(x) -^\sharp \mathsf{s}(y) \to x -^\sharp y \qquad \mathsf{s}(x) \div^\sharp \mathsf{s}(y) \to x -^\sharp y \qquad \mathsf{s}(x) \div^\sharp \mathsf{s}(y) \to (x - y) \div^\sharp \mathsf{s}(y)$$

Here $-^\sharp$ and $\div^\sharp$ are fresh function symbols. Consider the algebra $\mathcal{A}$ comprising polynomial interpretations over natural numbers:

$$\mathsf{0}_\mathcal{A} = 0 \qquad \mathsf{s}_\mathcal{A}(a) = a + 1 \qquad a -_\mathcal{A} b = a -^\sharp_\mathcal{A} b = a \div_\mathcal{A} b = a \div^\sharp_\mathcal{A} b = a$$

© Teppei Saito and Nao Hirokawa;
licensed under Creative Commons License CC-BY 4.0
9th International Conference on Formal Structures for Computation and Deduction (FSCD 2024).
Editor: Jakob Rehof; Article No. 13; pp. 13:1–13:20
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Under the interpretations, all rules in the TRS are weakly oriented, and all dependency pairs are strictly oriented. For instance, orientation of the last dependency pair is verified as follows: $\mathsf{s}_{\mathcal{A}}(a) \div^{\sharp}_{\mathcal{A}} \mathsf{s}_{\mathcal{A}}(b) = a+1 > a = (a -_{\mathcal{A}} b) \div^{\sharp}_{\mathcal{A}} \mathsf{s}_{\mathcal{A}}(b)$. Hence the termination is concluded by the dependency pair method.

Semantic labeling is a transformation method that labels function symbols in rewrite rules with values of their function arguments. Termination of the resulting TRS is equivalent to that of the original TRS.

▶ **Example 2** (continued from Example 1)**.** The termination of the TRS $\mathcal{R}$ can also be shown by semantic labeling. We use the same algebra $\mathcal{A}$ to label $-$ and $\div$ with values of their first arguments. The resulting labeled TRS consists of the rewrite rules

$$\begin{aligned} x -_a \mathsf{0} &\to x & \mathsf{0} \div_0 \mathsf{s}(y) &\to \mathsf{0} \\ \mathsf{s}(x) -_{a+1} \mathsf{s}(y) &\to x -_a y & \mathsf{s}(x) \div_{a+1} \mathsf{s}(y) &\to \mathsf{s}((x -_a y) \div_a \mathsf{s}(y)) \end{aligned}$$

for all $a \in \mathbb{N}$ and the auxiliary rules $x -_a y \to x -_b y$ and $x \div_a y \to x \div_b y$ for all $a, b \in \mathbb{N}$ with $a > b$. For instance, the label $a+1$ in $\mathsf{s}(x) \div_{a+1} \mathsf{s}(y)$ is the value of $\mathsf{s}(x)$ in $\mathcal{A}$ when x is assigned to a. The termination of this TRS is easily verified by the multiset path order with the (quasi-)precedence $\div_{a+1} \approx -_{a+1} \succ \div_a \approx -_a \succ \mathsf{s} \succ \mathsf{0}$ for all $a \in \mathbb{N}$.

In this paper we show that any termination proof by the dependency pair method can be effectively simulated by the combination of semantic labeling and a multiset path order. By incorporating a *flattening* technique (cf. [6]) in multiset path orders, this simulation result can be extended to Iborra et al.'s dependency pair method for relative termination [15]. Exploiting the simulation result, we improve applicability of this method.

An obstacle to the simulation results is a discrepancy between the two formalisms: the basic theorem of the dependency pair method is based on order pairs called *reduction pairs*, while Zantema's semantic labeling is based on well-founded algebras. We overcome this by reformulating semantic labeling and multiset path orders in forms suited for order pairs.

Interestingly, prior to the seminal paper [4], Arts [3] proved a restricted version of the dependency pair method by using Zantema's semantic labeling [29]. This is the first simulation result, and our work can be considered a revisit of the earlier attempt. This time we use Geser's generalized version [12]. Simulations by semantic labeling are not only of theoretical/historical interest but also of practical interest. In fact, based on the simulation result we relax a precondition of the result of Iborra et al. In addition, having proofs of the dependency pair method in a different route might ease formalization in proof assistants or extension to different rewrite formats.

The remaining part of the paper is organized as follows: In Section 2 we recall basic notions for term rewriting and multiset path orders based on order pairs. In order to simulate dependency pairs by Iborra et al. we introduce a variant of semantic labeling for relative termination in Section 3. In Section 4 we show how Arts and Giesl's dependency pair method can be simulated by the combination of semantic labeling and multiset path orders. In Section 5 we do the same for Iborra et al.'s dependency pair method, using a relative termination criterion that originates from multiset path orders and flattening. Correctness of the criterion is proved in Section 6. Exploiting this simulation result, we improve the applicability of the dependency pair method by Iborra et al. in Section 7. Section 8 concludes the paper by discussing experimental results and related work.

2 Preliminaries

Throughout the paper, we assume familiarity with term rewriting [5, 23].

Term Rewriting

Let $\mathcal{F}$ be a signature and $\mathcal{V}$ a countable set of variables with $\mathcal{F} \cap \mathcal{V} = \varnothing$. The set of all *terms* built from $\mathcal{F}$ and $\mathcal{V}$ is referred to as $\mathcal{T}(\mathcal{F}, \mathcal{V})$. When we need to indicate the arity of a function symbol f, we write $f^{(n)}$ for f. A term t is a *function application* if t is of the form $f(t_1, \ldots, t_n)$, and the *root symbol* f is denoted by $\mathsf{root}(t)$. The *size* $|t|$ of a term t is the number of function symbols and variables occurring in t. The set of function symbols or variables occurring in a term t is denoted by $\mathcal{F}\mathsf{un}(t)$ or $\mathcal{V}\mathsf{ar}(t)$, respectively.

Let $\square$ be a constant with $\square \notin \mathcal{F}$. *Contexts* are terms over $\mathcal{F} \cup \{\square\}$ that contain exactly one $\square$. The term resulting from replacing $\square$ in a context C by a term t is denoted by $C[t]$. We write $s \trianglerighteq t$ if there is a context C with $s = C[t]$. The strict part of $\trianglerighteq$ is denoted by $\triangleright$. A *substitution* is a mapping σ from variables to terms such that $\{x \in \mathcal{V} \mid \sigma(x) \neq x\}$ is finite. The application $t\sigma$ of a substitution σ to a term t is inductively defined as follows:

$$t\sigma = \begin{cases} \sigma(t) & \text{if } t \text{ is a variable} \\ f(t\sigma_1, \ldots, t\sigma_n) & \text{if } t = f(t_1, \ldots, t_n) \end{cases}$$

We say that a substitution σ is *grounding* for a set T of terms if $t\sigma$ is ground for all $t \in T$. The grounding target T may be omitted when T is clear from the context. A pair (ℓ, r) of terms is said to be a *rewrite rule* if ℓ is not a variable and every variable in r occurs in ℓ. Rewrite rules (ℓ, r) are written as $\ell \to r$. A set of rewrite rules is called a *term rewrite system* (TRS). Let $\mathcal{R}$ be a TRS. The relation $\to_\mathcal{R}$ is defined on terms as follows: $s \to_\mathcal{R} t$ if there exist a rewrite rule $\ell \to r \in \mathcal{R}$, a context C, and a substitution σ such that $s = C[\ell\sigma]$ and $t = C[r\sigma]$ hold. In particular, when $C = \square$ we may write $s \xrightarrow{\epsilon}_\mathcal{R} t$, which indicates that the rewriting happens at the root position. A term s is called a *normal form* with respect to a relation $\rightsquigarrow$ if there is no term t with $s \rightsquigarrow t$. The set of normal forms is denoted by $\mathsf{NF}(\rightsquigarrow)$. The TRS $\mathcal{R}$ is said to be *terminating* if $\to_\mathcal{R}$ is well-founded. *Relative termination* is a generalized notion of termination [11]. Given TRSs $\mathcal{R}$ and $\mathcal{S}$, we write $\to_{\mathcal{R}/\mathcal{S}}$ for the relation $\to_\mathcal{S}^* \cdot \to_\mathcal{R} \cdot \to_\mathcal{S}^*$. If $\to_{\mathcal{R}/\mathcal{S}}$ is well-founded, we say that $\mathcal{R}$ is *(relatively) terminating* with respect to $\mathcal{S}$ (or $\mathcal{R}/\mathcal{S}$ is terminating).

A pair $(\gtrsim, >)$ of a preorder and a strict order on the same set is called an *order pair* if $a > b$ holds whenever $a \gtrsim \cdot > \cdot \gtrsim b$. Here $>$ need not be the strict pair of $\gtrsim$. The order pair is *well-founded* if $>$ is well-founded. Relative termination is often shown by using well-founded order pairs on terms. A relation $\rightsquigarrow$ on terms is *closed under contexts* (or *monotone*) if $C[s] \rightsquigarrow C[t]$ holds whenever $s \rightsquigarrow t$ and C is a context, and it is *closed under substitutions* if $s\sigma \rightsquigarrow t\sigma$ holds whenever $s \rightsquigarrow t$ and σ is a substitution. We say $\rightsquigarrow$ has the *subterm property* if $s \rightsquigarrow t$ whenever $s \triangleright t$. A relation closed under contexts and substitutions is called a *rewrite relation*. A rewrite relation $\gtrsim$ is a *rewrite preorder* if it is a preorder. A rewrite relation $>$ is a *reduction order* if it is a well-founded order. Moreover, the pair $(\gtrsim, >)$ is called a *monotone reduction pair* if in addition they form an order pair. *Reduction pairs* $(\gtrsim, >)$ are akin to monotone reduction pairs, but the only difference is that $>$ may lack monotonicity.

▶ **Proposition 3.** *Let $\mathcal{R}, \mathcal{S}$ be TRSs. Then $\mathcal{R}/\mathcal{S}$ is terminating if and only if there exists a monotone reduction pair $(\gtrsim, >)$ such that $\mathcal{S} \subseteq {\gtrsim}$ and $\mathcal{R} \subseteq {>}$.* ◀

Ordered Algebras

Ordered algebras are key ingredients for constructing orders including ones for reduction pairs. An $\mathcal{F}$-*algebra* (or simply an algebra) is a pair $\mathcal{A} = (A, \{f_\mathcal{A}\}_{f \in \mathcal{F}})$, where A is a set called a *carrier*, and $f_\mathcal{A}$ is an n-ary function on A, called the *interpretation function* of a function symbol $f^{(n)} \in \mathcal{F}$. A mapping from $\mathcal{V}$ to A is called an *assignment* for $\mathcal{A}$. The interpretation $[\alpha]_\mathcal{A}(t)$ of a term t under an assignment α is inductively defined as follows:

$$[\alpha]_\mathcal{A}(t) = \begin{cases} \alpha(t) & \text{if } t \text{ is a variable} \\ f_\mathcal{A}([\alpha]_\mathcal{A}(t_1), \ldots, [\alpha]_\mathcal{A}(t_n)) & \text{if } t = f(t_1, \ldots, t_n) \end{cases}$$

In this paper we are interested in algebras equipped with *order pairs*. Let $\mathcal{A} = (A, \{f_\mathcal{A}\}_{f \in \mathcal{F}})$ be an algebra with A a non-empty set and $(\gtrsim, >)$ an order pair on A. The triple $(\mathcal{A}, \gtrsim, >)$ is called an *ordered algebra*. We say that the ordered algebra is

- *weakly monotone* if $f_\mathcal{A}(a_1, \ldots, a_i, \ldots, a_n) \gtrsim f_\mathcal{A}(a_1, \ldots, b, \ldots, a_n)$ for all $f^{(n)} \in \mathcal{F}$, argument positions $1 \leqslant i \leqslant n$, and $a_1, \ldots, a_n, b \in A$ with $a_i \gtrsim b$;
- *well-founded* if $>$ is well-founded.

Remark that monotonicity with respect to $>$ is not imposed on the interpretations $f_\mathcal{A}$. We write $s \gtrsim_\mathcal{A} t$ if $[\alpha]_\mathcal{A}(s) \gtrsim [\alpha]_\mathcal{A}(t)$ holds for all assignments α. Similarly, we write $s >_\mathcal{A} t$ if $[\alpha]_\mathcal{A}(s) > [\alpha]_\mathcal{A}(t)$ for all assignments α. The following facts are known:

- $(\gtrsim_\mathcal{A}, >_\mathcal{A})$ is an order pair and both $\gtrsim_\mathcal{A}$ and $>_\mathcal{A}$ are closed under substitutions;
- if $\mathcal{A}$ is weakly monotone then $\gtrsim_\mathcal{A}$ is closed under contexts; and
- if $\mathcal{A}$ is well-founded then $>_\mathcal{A}$ is well-founded.

Therefore, if $\mathcal{A}$ is weakly monotone and well-founded then $(\gtrsim_\mathcal{A}, >_\mathcal{A})$ is a reduction pair.

Multiset Path Orders

We use multiset path orders (MPOs) [8, Definition 5] based on *precedence pairs*, namely order pairs on the signature. The definition employs multiset extensions of order pairs [25] in a recursive way.

Let $(\gtrsim, >)$ be a pair of relations. For multisets X and Y we write $X \gtrsim_{\mathsf{mul}} Y$ if there are partitions $X = \{x_1, \ldots, x_n\} \uplus X'$ and $Y = \{y_1, \ldots, y_n\} \uplus Y'$ such that $x_i \gtrsim y_i$ for all $1 \leqslant i \leqslant n$, and for every $y \in Y'$ there exists $x \in X'$ with $x > y$. Furthermore, if in addition $X' \neq \varnothing$, we write $X >_{\mathsf{mul}} Y$. If $(\gtrsim, >)$ is an order pair, so is $(\gtrsim^{\mathsf{mul}}, >^{\mathsf{mul}})$. Moreover, if $>$ is well-founded, so is $>^{\mathsf{mul}}$.

▶ **Definition 4.** *Let $(\succsim, \succ)$ be a precedence pair on $\mathcal{F}$. The order pair $(\succsim_{\mathsf{mpo}}, \succ_{\mathsf{mpo}})$ of the* multiset path orders *is inductively defined on terms over $\mathcal{F}$ as follows:*

- *$s \succ_{\mathsf{mpo}} t$ if $s = f(s_1, \ldots, s_m)$ and one of the following conditions holds.*
  1. *$s_i \succsim_{\mathsf{mpo}} t$ for some $1 \leqslant i \leqslant m$.*
  2. *$t = g(t_1, \ldots, t_n)$, $f \succ g$, and $s \succ_{\mathsf{mpo}} t_j$ for all $1 \leqslant j \leqslant n$.*
  3. *$t = g(t_1, \ldots, t_n)$, $f \succsim g$, and $\{s_1, \ldots, s_m\} \succ_{\mathsf{mpo}}^{\mathsf{mul}} \{t_1, \ldots, t_n\}$.*
- *$s \succsim_{\mathsf{mpo}} t$ if either s and t are the same variable, or $s = f(s_1, \ldots, s_m)$ and one of the following conditions holds.*
  1. *$s_i \succsim_{\mathsf{mpo}} t$ for some $1 \leqslant i \leqslant m$.*
  2. *$t = g(t_1, \ldots, t_n)$, $f \succ g$, and $s \succ_{\mathsf{mpo}} t_j$ for all $1 \leqslant j \leqslant n$.*
  3. *$t = g(t_1, \ldots, t_n)$, $f \succsim g$, and $\{s_1, \ldots, s_m\} \succsim_{\mathsf{mpo}}^{\mathsf{mul}} \{t_1, \ldots, t_n\}$.*

Here $(\succsim_{\mathsf{mpo}}^{\mathsf{mul}}, \succ_{\mathsf{mpo}}^{\mathsf{mul}})$ stands for the multiset extension of $(\succsim_{\mathsf{mpo}}, \succ_{\mathsf{mpo}})$.

A small remark is that the definition above is based on mutual recursion (cf. [26, Definition 4]). Basic properties of MPOs are readily proved.

▶ **Theorem 5.** *For every well-founded precedence pair the induced order pair $(\succsim_{\mathsf{mpo}}, \succ_{\mathsf{mpo}})$ is a monotone reduction pair.* ◀

3 Semantic Labeling for Relative Termination

We introduce semantic labeling for *relative termination*. As stated in the introduction, the original version of semantic labeling [29, Theorem 8] (see also [23, Section 6.5.4]) employs a well-founded algebra and labeling functions to reduce termination of a given TRS into termination of the labeled TRS. Using the notion of relative rewriting, Geser [12] got rid of the well-foundedness requirement from employed ordered algebras. Our variant of semantic labeling is a straightforward adaptation of his result to our setting.

Let $\mathcal{A} = (A, \{f_\mathcal{A}\}_{f\in\mathcal{F}})$ be an algebra, $\{L_f\}_{f\in\mathcal{F}}$ a family of non-empty subsets of A, and $\{\mathsf{lab}_f\}_{f\in\mathcal{F}}$ a family of functions where lab_f is a mapping from A^n to L_f for each $f^{(n)} \in \mathcal{F}$. The pair $(\{L_f\}_{f\in\mathcal{F}}, \{\mathsf{lab}_f\}_{f\in\mathcal{F}})$ (denoted by $\mathcal{L}$ in this paper) is called a *labeling* for $\mathcal{A}$. Elements in L_f are called *labels*. For each $f^{(n)} \in \mathcal{F}$ and $a \in L_f$ we introduce a fresh function symbol f_a if $|L_f| > 1$, and if $L_f = \{a\}$ we reuse the original symbol, namely define $f_a = f$. The *labeled signature* $\{f_a^{(n)} \mid f^{(n)} \in \mathcal{F}$ and $a \in L_f\}$ is denoted by $\mathcal{F}_\mathsf{lab}$. Note that, in our formulation, symbols f not subject to labeling (i.e., $|L_f| = 1$) are still included in $\mathcal{F}_\mathsf{lab}$. The *labeling function* lab for terms $t \in \mathcal{T}(\mathcal{F}, \mathcal{V})$ under an assignment $\alpha : \mathcal{V} \to A$ is defined as

$$\mathsf{lab}(t, \alpha) = \begin{cases} t & \text{if } t \text{ is a variable} \\ f_a(\mathsf{lab}(t_1, \alpha), \ldots, \mathsf{lab}(t_n, \alpha)) & \text{if } t = f(t_1, \ldots, t_n) \end{cases}$$

where in the second case $a = \mathsf{lab}_f([\alpha]_\mathcal{A}(t_1), \ldots, [\alpha]_\mathcal{A}(t_n))$. The resulting term $\mathsf{lab}(\alpha, t)$ is a term over $\mathcal{F}_\mathsf{lab}$. Let $\mathcal{R}$ be a TRS over $\mathcal{F}$. The *labeled TRS* $\mathcal{R}_\mathsf{lab}$ is defined as follows:

$$\mathcal{R}_\mathsf{lab} = \{\mathsf{lab}(\ell, \alpha) \to \mathsf{lab}(r, \alpha) \mid \ell \to r \in \mathcal{R} \text{ and } \alpha \text{ is an assignment}\}$$

▶ **Example 6.** Consider the one-rule TRS $\mathcal{R} = \{\mathsf{f}(\mathsf{f}(x)) \to \mathsf{f}(\mathsf{g}(\mathsf{f}(x)))\}$ and also consider the following algebra $\mathcal{A} = (\{0, 1\}, \{\mathsf{f}_\mathcal{A}, \mathsf{g}_\mathcal{A}\})$ and labeling $\mathcal{L} = (\{L_\mathsf{f}, L_\mathsf{g}\}, \{\mathsf{lab}_\mathsf{f}, \mathsf{lab}_\mathsf{g}\})$:

$$\mathsf{f}_\mathcal{A}(x) = 1,\ \mathsf{g}_\mathcal{A}(x) = 0 \qquad L_\mathsf{f} = \{0, 1\},\ L_\mathsf{g} = \{0\} \qquad \mathsf{lab}_\mathsf{f}(x) = x,\ \mathsf{lab}_\mathsf{g}(x) = x$$

The labeling results in the TRS $\mathcal{R}_\mathsf{lab} = \{\mathsf{f}_1(\mathsf{f}_0(x)) \to \mathsf{f}_0(\mathsf{g}(\mathsf{f}_0(x))),\ \mathsf{f}_1(\mathsf{f}_1(x)) \to \mathsf{f}_0(\mathsf{g}(\mathsf{f}_1(x)))\}$.

Our variant of semantic labeling employs weakly monotone algebras and weakly monotone labelings. Let $(\mathcal{A}, \gtrsim, >)$ be an ordered algebra. We say that a labeling $(\{L_f\}_{f\in\mathcal{F}}, \{\mathsf{lab}_f\}_{f\in\mathcal{F}})$ is *weakly monotone* if $\mathsf{lab}_f(a_1, \ldots, a_i, \ldots, a_n) \gtrsim \mathsf{lab}_f(a_1, \ldots, b, \ldots, a_n)$ for all $f^{(n)} \in \mathcal{F}$, argument positions $1 \leqslant i \leqslant n$, and $a_1, \ldots, a_n, b \in A$ with $a_i \gtrsim b$. We define the TRS of *decreasing rules* with respect to a binary relation $\rightsquigarrow$ on L as follows:

$$\mathcal{D}\mathsf{ec}(\rightsquigarrow) = \{f_a(x_1, \ldots, x_n) \to f_b(x_1, \ldots, x_n) \mid f^{(n)} \in \mathcal{F} \text{ and } a, b \in L_f \text{ with } a \rightsquigarrow b\}$$

Here $x_1, \ldots, x_n$ are pairwise different variables.

We are ready to state the main theorem of semantic labeling for relative termination. The theorem speaks about weakly monotone algebras $(\mathcal{A}, \gtrsim, >)$ but actually their strict orders $>$ are irrelevant for this theorem. In other words, the theorem holds regardless of how $>$ is like. Therefore, for brevity we may write $(\mathcal{A}, \gtrsim)$ instead of $(\mathcal{A}, \gtrsim, >)$. The proof is found in Appendix A.

▶ **Theorem 7.** *Let $\mathcal{R}$ and $\mathcal{S}$ be TRSs and $(\mathcal{A}, \gtrsim)$ a weakly monotone algebra with $\mathcal{R} \cup \mathcal{S} \subseteq {\gtrsim_\mathcal{A}}$, and let $\mathcal{L}$ be a weakly monotone labeling for $(\mathcal{A}, \gtrsim)$. Then $\mathcal{R}/\mathcal{S}$ is terminating if and only if $\mathcal{R}_\mathsf{lab}/(\mathcal{S}_\mathsf{lab} \cup \mathcal{D}\mathsf{ec}(\gtrsim))$ is terminating.*

Note that until Section 4 we only use the theorem with $\mathcal{S} = \varnothing$, which coincides with Geser's semantic labeling [12, Corollary 1]. With a small example we illustrate a termination proof based on the theorem.

▶ **Example 8** (continued from Example 6)**.** Let $\gtrsim$ be the quasi-order on $\{0,1\}$ with $1 \gtrsim 0$. Then $(\mathcal{A}, \gtrsim)$ and $\mathcal{L}$ are weakly monotone. Since the inequality $\mathsf{f}_\mathcal{A}(\mathsf{f}_\mathcal{A}(a)) = 1 \gtrsim 1 = \mathsf{f}_\mathcal{A}(\mathsf{g}_\mathcal{A}(\mathsf{f}_\mathcal{A}(a)))$ holds for all $a \in \{0,1\}$, the inclusion $\mathcal{R} \subseteq {\gtrsim_\mathcal{A}}$ follows. The TRS $\mathcal{D}\mathsf{ec}(\gtrsim)$ consists of the four rules: $\mathcal{D}\mathsf{ec}(\gtrsim) = \{\mathsf{f}_0(x) \to \mathsf{f}_0(x), \mathsf{f}_1(x) \to \mathsf{f}_0(x), \mathsf{f}_1(x) \to \mathsf{f}_1(x), \mathsf{g}(x) \to \mathsf{g}(x)\}$. By taking the MPO with the precedence $\mathsf{f}_1 \succ \mathsf{f}_0 \succ \mathsf{g}$ we obtain the inclusions $\mathcal{R}_\mathsf{lab} \subseteq {\succ_\mathsf{mpo}}$ and $\mathcal{D}\mathsf{ec}(\gtrsim) \subseteq {\succsim_\mathsf{mpo}}$. Therefore, $\mathcal{R}_\mathsf{lab}/\mathcal{D}\mathsf{ec}(\gtrsim)$ is terminating by Theorem 5. Hence, by applying Theorem 7 we conclude termination of $\mathcal{R}$.

The original statement of (quasi-model based) semantic labeling [29, Theorem 8] can be seen as a special case of Theorem 7, which relies on termination rather than relative termination. To see this, recall that a term rewrite system $\mathcal{R} \cup \mathcal{S}$ is terminating if and only if $\mathcal{R}/\mathcal{S}$ and $\mathcal{S}$ are terminating [11].

▶ **Corollary 9** ([29, Theorem 8])**.** *Let $(\mathcal{A}, \geqslant)$ be a weakly monotone well-founded algebra with $\geqslant$ a partial order, $\mathcal{L}$ a weakly monotone labeling for $(\mathcal{A}, \geqslant)$, and $\mathcal{R}$ a TRS with $\mathcal{R} \subseteq {\geqslant_\mathcal{A}}$. Then $\mathcal{R}$ is terminating if and only if $\mathcal{R}_\mathsf{lab} \cup \mathcal{D}\mathsf{ec}(>)$ is terminating.*

Proof. By Theorem 7 termination of $\mathcal{R}$ is equivalent to that of $\mathcal{R}_\mathsf{lab}/\mathcal{D}\mathsf{ec}(\geqslant)$. Because $\to_{\mathcal{D}\mathsf{ec}(\geqslant)}$ and $\to^=_{\mathcal{D}\mathsf{ec}(>)}$ coincide, the latter is equivalent to termination of $\mathcal{R}_\mathsf{lab}/\mathcal{D}\mathsf{ec}(>)$. Since $>$ is well-founded, $\mathcal{D}\mathsf{ec}(>)$ is terminating. Therefore, $\mathcal{R}_\mathsf{lab}/\mathcal{D}\mathsf{ec}(>)$ is terminating if and only if $\mathcal{R}_\mathsf{lab} \cup \mathcal{D}\mathsf{ec}(>)$ is terminating. ◀

4 Simulating Dependency Pairs for Termination

We recall a basic form of Arts and Giesl's dependency pair method. Let $\mathcal{G}$ be a subset of the signature $\mathcal{F}$. Given an n-ary function symbol f in $\mathcal{G}$, we introduce a fresh n-ary function symbol $f^\sharp$ called a *marked symbol*. The set of marked symbols is denoted by $\mathcal{G}^\sharp$. Given a term $t = f(t_1, \ldots, t_n)$, we write $t^\sharp$ for the term $f^\sharp(t_1, \ldots, t_n)$. For a TRS $\mathcal{R}$ the set $\mathcal{D}_\mathcal{R}$ of *defined symbols* are defined by $\mathcal{D}_\mathcal{R} = \{f \mid f(t_1, \ldots, t_n) \to r \in \mathcal{R}\}$. The difference $\mathcal{F} \setminus \mathcal{D}_\mathcal{R}$ is denoted by $\mathcal{C}_\mathcal{R}$, and the symbols in $\mathcal{C}_\mathcal{R}$ are called *constructor symbols* or just constructors.

▶ **Definition 10.** *Let $\mathcal{R}$ be a TRS over the signature $\mathcal{F}$ and let $\mathcal{G} \subseteq \mathcal{F}$. The TRS $\mathsf{DP}_\mathcal{G}(\mathcal{R})$ over $\mathcal{F} \cup \mathcal{G}^\sharp$ is defined by $\mathsf{DP}_\mathcal{G}(\mathcal{R}) = \{\ell^\sharp \to t^\sharp \mid \ell \to r \in \mathcal{R},\ r \trianglerighteq t,\ \mathsf{root}(\ell), \mathsf{root}(t) \in \mathcal{G}, \text{ and } \ell \ntriangleright t\}$. The TRS $\mathsf{DP}_{\mathcal{D}_\mathcal{R}}(\mathcal{R})$ is abbreviated to $\mathsf{DP}(\mathcal{R})$, and its rules are called* dependency pairs *of $\mathcal{R}$.*

Note that the non-subterm condition $\ell \ntriangleright t$ is due to Dershowitz [9], and it was not included in the original definition of $\mathsf{DP}(\mathcal{R})$ [4].

▶ **Theorem 11** ([4, 13])**.** *A TRS $\mathcal{R}$ is terminating if and only if $\mathcal{R} \subseteq {\gtrsim}$ and $\mathsf{DP}(\mathcal{R}) \subseteq {>}$ for some reduction pair $(\gtrsim, >)$.*

▶ Remark 12. Today the termination condition in the theorem is stated as *finiteness* of $(\mathsf{DP}(\mathcal{R}), \mathcal{R})$; see [13] for the definition. Whenever $(\mathcal{P}, \mathcal{R})$ is finite, the relation $\to^*_\mathcal{R}$ and the restriction of $\to^+_{\mathcal{P}/\mathcal{R}}$ to $\mathcal{R}$-terminating terms form a reduction pair; the restriction takes care of the so-called *minimality* condition of chains. So finiteness and existence of a suitable reduction pair are equivalent.

For any weakly monotone well-founded algebra $(\mathcal{A}, \gtrsim, >)$ the induced order pair $(\gtrsim_\mathcal{A}, >_\mathcal{A})$ forms a reduction pair. Conversely, for every reduction pair $(\gtrsim, >)$ the ordered algebra $(\mathcal{A}, \gtrsim, >)$ fulfills the desired properties where $\mathcal{A}$ is the *term algebra* $(\mathcal{T}(\mathcal{F} \cup \mathcal{D}^\sharp_\mathcal{R}, \mathcal{V}), \{f_\mathcal{A}\}_{f \in \mathcal{F}})$ defined by $f_\mathcal{A}(t_1, \ldots, t_n) = f(t_1, \ldots, t_n)$. Therefore, in the remaining part of the paper we investigate Corollary 13 below instead of Theorem 11.

▶ **Corollary 13.** *A TRS $\mathcal{R}$ is terminating if and only if $\mathcal{R} \subseteq \gtrsim_\mathcal{A}$ and $\mathsf{DP}(\mathcal{R}) \subseteq >_\mathcal{A}$ for some weakly monotone well-founded $(\mathcal{F} \cup \mathcal{D}^\sharp_\mathcal{R})$-algebra $(\mathcal{A}, \gtrsim, >)$.* ◀

▶ **Example 14.** Example 1 is an example of termination proofs by Corollary 13. The algebra $\mathcal{A}$ uses the standard orders on $\mathbb{N}$, and its interpretations are weakly monotone. Therefore $(\geqslant_\mathcal{A}, >_\mathcal{A})$ is a reduction pair. Since $\mathcal{R} \subseteq \geqslant_\mathcal{A}$ and $\mathsf{DP}(\mathcal{R}) \subseteq >_\mathcal{A}$ we conclude the termination by Corollary 13.

We exemplify how proofs based on the dependency pair method can be simulated by semantic labeling with MPOs, introducing a few necessary definitions.

▶ **Definition 15.** *Let $\mathcal{R}$ be a TRS over the signature $\mathcal{F}$ and $(\mathcal{A}, \gtrsim, >)$ an ordered $(\mathcal{F} \cup \mathcal{D}^\sharp_\mathcal{R})$-algebra on a carrier A. Fix an arbitrary element $\bullet \in A$. We define the* $\mathcal{A}$-induced labeling *$\mathcal{L}_\mathcal{A} = (\{L_f\}_{f \in \mathcal{F}}, \{\mathsf{lab}_f\}_{f \in \mathcal{F}})$ as follows:*

$$L_f = \begin{cases} A & \text{if } f \in \mathcal{D}_\mathcal{R} \\ \{\bullet\} & \text{otherwise} \end{cases} \qquad \mathsf{lab}_f(a_1, \ldots, a_n) = \begin{cases} f^\sharp_\mathcal{A}(a_1, \ldots, a_n) & \text{if } f \in \mathcal{D}_\mathcal{R} \\ \bullet & \text{otherwise} \end{cases}$$

We also define the precedence pair $(\succsim, \succ)$, called $\mathcal{A}$-induced precedence pair*, on the labeled signature as follows:*

- *$f_a \succsim g_b$ if either $f, g \in \mathcal{D}_\mathcal{R}$ and $a \gtrsim b$, or $g \in \mathcal{C}_\mathcal{R}$*
- *$f_a \succ g_b$ if either $f, g \in \mathcal{D}_\mathcal{R}$ and $a > b$, or $f \in \mathcal{D}_\mathcal{R}$ and $g \in \mathcal{C}_\mathcal{R}$*

So in the precedence pair, constructors are smaller than (labeled) defined symbols.

▶ **Example 16** (continued from Example 14, see also Example 2)**.** The $\mathcal{A}$-induced labeling gives the labeled TRS $\mathcal{R}_\mathsf{lab}$ consisting of the rules

$$x -_a \mathsf{0} \to x \qquad \mathsf{0} \div_0 \mathsf{s}(y) \to \mathsf{0}$$
$$\mathsf{s}(x) -_{a+1} \mathsf{s}(y) \to x -_a y \qquad \mathsf{s}(x) \div_{a+1} \mathsf{s}(y) \to \mathsf{s}((x - y) \div_a \mathsf{s}(y))$$

for all $a \in \mathbb{N}$. The TRS $\mathcal{D}\mathsf{ec}(\geqslant)$ is also the infinite set consisting of the rules

$$x -_a y \to x -_b y \qquad x \div_a y \to x \div_b y$$

for all $a, b \in \mathbb{N}$ with $a \geqslant b$. By Theorem 7 the termination of $\mathcal{R}$ follows if we show that of $\mathcal{R}_\mathsf{lab}/\mathcal{D}\mathsf{ec}(\geqslant)$. The $\mathcal{A}$-induced precedence pair $(\succsim, \succ)$ satisfies $\div_{a+1} \approx -_{a+1} \succ \div_a \approx -_a \succ \mathsf{s} \succ \mathsf{0}$ for all $a \in \mathbb{N}$. Here $f \approx g$ stands for $f \succsim g$ and $g \succsim f$. It is easy to see $\mathcal{R}_\mathsf{lab} \subseteq \succ_\mathsf{mpo}$ and $\mathcal{D}\mathsf{ec}(\geqslant) \subseteq \succsim_\mathsf{mpo}$. Hence, $\mathcal{R}$ is terminating by Theorem 5.

Although a multiset path order is used in the last example, other path orders such as lexicographic path orders (LPOs) [17] can also be used for showing termination of $\mathcal{R}_\mathsf{lab}/\mathcal{D}\mathsf{ec}(\gtrsim)$. In order to manifest this fact, we introduce a minimalistic termination criterion, inspired by precedence termination (cf. [20, Lemma 1]).

▶ **Definition 17.** *Let $(\succsim, \succ)$ be a precedence pair and $\mathcal{G} \subseteq \mathcal{F}$. The relation $\succ_\mathcal{G}$ on terms is inductively defined as follows: $s \succ_\mathcal{G} t$ if $s = f(s_1, \ldots, s_m)$, $f \in \mathcal{F} \setminus \mathcal{G}$, and one of the following two conditions holds.*

(1) $s \rhd t$.
(2) $t = g(t_1, \ldots, t_n)$, $f \succ g$, and $s \succ_{\mathcal{G}} t_j$ for all $1 \leqslant j \leqslant n$.
The relation $\succsim_{\mathcal{G}}$ *on terms is defined as follows:* $s \succsim_{\mathcal{G}} t$ *if* $s = f(t_1, \ldots, t_n)$, $t = g(t_1, \ldots, t_n)$, $f \in \mathcal{F} \setminus \mathcal{G}$, *and* $f \succsim g$.

Due to the minimalistic definition, rules like commutativity $f(x, y) \to f(y, x)$ cannot be ordered by $\succsim_{\mathcal{G}}$.

▶ **Lemma 18.** *If* $s \succ_{\mathcal{G}} t$ *or* $s \succsim_{\mathcal{G}} t$ *then* $s \succ_{\mathsf{mpo}} t$ *or* $s \succsim_{\mathsf{mpo}} t$, *respectively.* ◀

In general, $\succ_{\mathcal{G}}$ and $\succsim_{\mathcal{G}}$ do not form a monotone reduction pair. However, they give the following simple criterion for relative termination.

▶ **Proposition 19.** *Let* $(\succsim, \succ)$ *be a well-founded precedence pair. Then* $\mathcal{R}/\mathcal{S}$ *is terminating if there exists a subset* $\mathcal{G}$ *of* $\mathcal{F}$ *such that* $\mathcal{R} \subseteq \succ_{\mathcal{G}}$ *and* $\mathcal{S} \subseteq \succsim_{\mathcal{G}}$.

Proof. By Lemma 18 and Theorem 5. ◀

We establish the main result of this section, using Proposition 19 with $\mathcal{G} = \mathcal{C}_{\mathcal{R}}$. Note that $\mathcal{C}_{\mathcal{R}}$ is included in $\mathcal{F}_{\mathsf{lab}}$ because $\mathcal{L}_{\mathcal{A}}$ do not label constructor symbols of $\mathcal{R}$.

▶ **Lemma 20.** *Suppose* $\mathsf{DP}(\mathcal{R}) \subseteq >_{\mathcal{A}}$ *and* $\mathcal{G} = \mathcal{C}_{\mathcal{R}}$, *and consider the* $\mathcal{A}$*-induced labeling and the* $\mathcal{A}$*-induced precedence* $(\succsim, \succ)$. *If* $\ell \to r \in \mathcal{R}$ *then* $\mathsf{lab}(\ell, \alpha) \succ_{\mathcal{G}} \mathsf{lab}(t, \alpha)$ *for all subterms* t *of* r *and assignments* α.

Proof. Suppose $\ell \to r \in \mathcal{R}$ and $r \unrhd t$. Let α be an assignment. We show $\mathsf{lab}(\ell, \alpha) \succ_{\mathcal{G}} \mathsf{lab}(t, \alpha)$ by structural induction on t. Because $\ell \to r$ is a rewrite rule, ℓ must be of form $f(\ell_1, \ldots, \ell_m)$ with $f \in \mathcal{D}_{\mathcal{R}}$. If $\ell \rhd t$ then $\mathsf{lab}(\ell, \alpha) \rhd \mathsf{lab}(t, \alpha)$ and thereby $\mathsf{lab}(\ell, \alpha) \succ_{\mathcal{G}} \mathsf{lab}(t, \alpha)$. Otherwise, t is not a variable because $\ell \rhd t$ follows from $t \in \mathcal{V}\mathsf{ar}(r) \subseteq \mathcal{V}\mathsf{ar}(\ell)$ and $\ell \notin \mathcal{V}$. So suppose $t = g(t_1, \ldots, t_n)$. By the induction hypothesis $\mathsf{lab}(\ell, \alpha) \succ_{\mathcal{G}} \mathsf{lab}(t_j, \alpha)$ for all $j \in \{1, \ldots, n\}$. We have $\mathsf{lab}(\ell, \alpha) = f_a(\mathsf{lab}(\ell_1, \alpha), \ldots, \mathsf{lab}(\ell_m, \alpha))$ where $a = f^\sharp_{\mathcal{A}}([\alpha]_{\mathcal{A}}(\ell_1), \ldots, [\alpha]_{\mathcal{A}}(\ell_m))$. We distinguish two cases, depending on g.

- If $g \notin \mathcal{D}_{\mathcal{R}}$ then $f_a \succ g$. Therefore $\mathsf{lab}(\ell, \alpha) \succ_{\mathcal{G}} g(\mathsf{lab}(t_1, \alpha), \ldots, \mathsf{lab}(t_n, \alpha)) = \mathsf{lab}(t, \alpha)$.
- If $g \in \mathcal{D}_{\mathcal{R}}$ then $\ell^\sharp \to t^\sharp \in \mathsf{DP}(\mathcal{R})$ because of $\ell \ntrianglerighteq t$. From $\mathsf{DP}(\mathcal{R}) \subseteq >_{\mathcal{A}}$ we obtain $\ell^\sharp >_{\mathcal{A}} t^\sharp$. So by the definition of $>_{\mathcal{A}}$ we obtain $a > b$ for $b = g^\sharp_{\mathcal{A}}([\alpha]_{\mathcal{A}}(t_1), \ldots, [\alpha]_{\mathcal{A}}(t_n))$, and thus $f_a \succ g_b$ follows. Therefore $\mathsf{lab}(\ell, \alpha) \succ_{\mathcal{G}} g_b(\mathsf{lab}(t_1, \alpha), \ldots, \mathsf{lab}(t_n, \alpha)) = \mathsf{lab}(t, \alpha)$. ◀

▶ **Theorem 21.** *Let* $\mathcal{R}$ *be a TRS and* $(\mathcal{A}, \gtrsim, >)$ *a weakly monotone well-founded algebra with* $\mathcal{R} \subseteq \gtrsim_{\mathcal{A}}$. *The following statements hold for* $\mathcal{G} = \mathcal{C}_{\mathcal{R}}$.

1. *The* $\mathcal{A}$*-induced labeling is a weakly monotone labeling for* $(\mathcal{A}, \gtrsim)$.
2. *The* $\mathcal{A}$*-induced precedence pair* $(\succsim, \succ)$ *is well-founded.*
3. *If* $\mathsf{DP}(\mathcal{R}) \subseteq >_{\mathcal{A}}$ *then* $\mathcal{R}_{\mathsf{lab}} \subseteq \succ_{\mathcal{G}}$.
4. $\mathcal{D}\mathsf{ec}(\gtrsim) \subseteq \succsim_{\mathcal{G}}$.

Proof. The third claim follows from Lemma 20. The other claims are straightforward. ◀

Theorem 21 states that, given any termination proof by the basic dependency pair method (Corollary 13), one can construct a corresponding termination proof by semantic labelling (Theorem 7) and precedence-based termination (Proposition 19). This simulation result is conceivable as an alternative proof for Theorem 11. While the standard correctness proof of the dependency pair method relies on the notion of minimal non-terminating term [4], the one via semantic labeling directly captures the decreasing measure (i.e., labels of defined symbols) by recursive path orders such as MPOs and LPOs.

We conclude the section by stating why we adopted Geser's version of semantic labeling. Since the original semantic labeling (Corollary 9) employs a weakly monotone well-founded algebra, from a given reduction pair $(\gtrsim, >)$ we need to construct a single well-founded partial order that plays *both* roles of $\gtrsim$ and $>$. Geser's version resolves this discrepancy, hiding $>$ behind the relative termination condition of labeled systems.

5 Simulating Dependency Pairs for Relative Termination

Iborra et al. [15] developed a natural extension of the dependency pair method to relative termination. We show that this extension can also be simulated by semantic labeling. First we recall their main theorem.

▶ **Definition 22.** *Let $\mathcal{R}$ and $\mathcal{S}$ be TRSs. We say that $\mathcal{R}$* dominates *$\mathcal{S}$ if $\mathcal{F}\mathsf{un}(r) \cap \mathcal{D}_{\mathcal{R}} = \varnothing$ for all $\ell \to r \in \mathcal{S}$. Let $|t|_x$ denote the number of occurrences of a variable x in a term t. A pair (ℓ, r) of terms is called* non-duplicating *if $|\ell|_x \geqslant |r|_x$ for all variables x, and a rule $\ell \to r$ is* non-duplicating *if (ℓ, r) is so. Finally, a TRS $\mathcal{R}$ is* non-duplicating *if every rule in $\mathcal{R}$ is non-duplicating.*

▶ **Theorem 23** ([15, Theorem 2])**.** *Suppose that a TRS $\mathcal{R}$ dominates a non-duplicating TRS $\mathcal{S}$. Then $\mathcal{R}/\mathcal{S}$ is terminating if and only if $\mathsf{DP}(\mathcal{R}) \subseteq >_{\mathcal{A}}$ and $\mathcal{R} \cup \mathcal{S} \subseteq \gtrsim_{\mathcal{A}}$ for some weakly monotone well-founded algebra $(\mathcal{A}, \gtrsim, >)$.*

Theorem 23 is a generalization of the basic dependency pair method (Corollary 13), since the empty TRS is non-duplicating and dominated by any TRS.

▶ **Example 24.** Recall the TRS $\mathcal{R}$ of division from Example 1. We show the relative termination of $\mathcal{R}$ with respect to the TRS $\mathcal{S} = \{\mathsf{rand}(x) \to x, \mathsf{rand}(x) \to \mathsf{rand}(\mathsf{s}(x))\}$. Since the TRS $\mathcal{S}$ is non-duplicating and $\mathcal{R}$ dominates $\mathcal{S}$, we may use Theorem 23 to show termination of $\mathcal{R}/\mathcal{S}$. The set $\mathsf{DP}(\mathcal{R})$ consists of the three rules, see Example 1. Let $(\mathcal{A}, \geqslant, >)$ be the weakly monotone well-founded algebra, where the carrier consists of ordinal numbers below ω^2 and the interpretations are given by the equations:

$$\mathsf{0}_{\mathcal{A}} = 0 \quad \mathsf{s}_{\mathcal{A}}(a) = a + 1 \quad \mathsf{rand}_{\mathcal{A}}(a) = a + \omega \quad a -_{\mathcal{A}} b = a -^{\sharp}_{\mathcal{A}} b = a \div_{\mathcal{A}} b = a \div^{\sharp} b = a$$

It is easy to verify $\mathcal{R} \cup \mathcal{S} \subseteq \geqslant_{\mathcal{A}}$ and $\mathsf{DP}(\mathcal{R}) \subseteq >_{\mathcal{A}}$. For instance, the last rules in $\mathsf{DP}(\mathcal{R})$ and $\mathcal{S}$ are oriented as the inequalities

$$\begin{aligned} \mathsf{s}_{\mathcal{A}}(a) \div^{\sharp}_{\mathcal{A}} \mathsf{s}_{\mathcal{A}}(b) &= a + 1 > a = (a -_{\mathcal{A}} b) \div^{\sharp}_{\mathcal{A}} \mathsf{s}_{\mathcal{A}}(b) \\ \mathsf{rand}_{\mathcal{A}}(a) &= a + \omega = a + 1 + \omega = \mathsf{rand}_{\mathcal{A}}(\mathsf{s}_{\mathcal{A}}(a)) \end{aligned}$$

hold for all ordinals $a, b < \omega^2$. Hence, $\mathcal{R}$ is terminating.

For showing an analog of Theorem 21 in a relative termination setting, from a given reduction pair $(\geqslant_{\mathcal{A}}, >_{\mathcal{A}})$ we construct the $\mathcal{A}$-induced labeling and precedence in the same way. However, existing syntactical termination methods, such as precedence-based termination (Proposition 19) and MPOs, are still incapable of showing termination of labeled systems due to problematic rules in relative systems like $\mathsf{rand}(x) \to \mathsf{rand}(\mathsf{s}(x))$.

▶ **Example 25** (continued from Example 24)**.** Following the construction of Theorem 21, we obtain the TRS $\mathcal{R}_{\mathsf{lab}}$ consisting of

$$\begin{aligned} x -_a \mathsf{0} &\to x & \mathsf{0} \div_0 \mathsf{s}(y) &\to \mathsf{0} \\ \mathsf{s}(x) -_{a+1} \mathsf{s}(y) &\to x -_a y & \mathsf{s}(x) \div_{a+1} \mathsf{s}(y) &\to \mathsf{s}((x - y) \div_a \mathsf{s}(y)) \end{aligned}$$

for all ordinals $a < \omega^2$. The TRS $\mathcal{S}_{\mathsf{lab}}$ is the same as $\mathcal{S}$ since we do not label constructors, and the TRS $\mathcal{D}\mathsf{ec}(\geqslant)$ consists of

$$x -_a y \to x -_b y \qquad\qquad x \div_a y \to x \div_b y$$

for all ordinals $a, b < \omega^2$ with $a \geqslant b$. The inclusions $\mathcal{R}_{\mathsf{lab}} \subseteq \succ_{\mathsf{mpo}}$ and $\mathcal{D}\mathsf{ec}(\geqslant) \subseteq \succsim_{\mathsf{mpo}}$ hold, but $\mathcal{S}_{\mathsf{lab}} \subseteq \succsim_{\mathsf{mpo}}$ does not. Actually, any monotone reduction pair $(\gtrsim, >)$ satisfying the subterm property $\rhd \subseteq >$ is unable to orient $\mathcal{S}_{\mathsf{lab}}$.

We overcome this problem by *flattening* inspired by [6] and [15, Definition 3]. For each $k \in \mathbb{N}$ we introduce a fresh k-ary function symbol c_k, called *compound symbols*. The set of all compound symbols is referred to as $\mathcal{F}_{\mathsf{c}}$.

▶ **Definition 26.** *Let $\mathcal{G}$ be a set of function symbols in $\mathcal{F}$. The* flattening TRS $\mathsf{F}(\mathcal{G})$ *over the signature $\mathcal{F} \cup \mathcal{F}_{\mathsf{c}}$ consists of the rules $f(x_1, \ldots, x_n) \to \mathsf{c}_n(x_1, \ldots, x_n)$, $\mathsf{c}_1(x) \to x$ and*

$$\mathsf{c}_{k+n+1}(x_1, \ldots, x_k, \mathsf{c}_m(y_1, \ldots, y_m), z_1, \ldots, z_n) \to \mathsf{c}_{k+m+n}(x_1, \ldots, x_k, y_1, \ldots, y_m, z_1, \ldots, z_n)$$

for all $k, m, n \in \mathbb{N}$ and $f^{(n)} \in \mathcal{G}$. Since the TRS is terminating and confluent, every term t admits exactly one normal form, which we denote by $t{\downarrow_{\mathcal{G}}}$. Such a normal form is called a flattened term*. We abbreviate $t{\downarrow_{\mathcal{G}}}$ to $t{\downarrow}$ whenever $\mathcal{G}$ is clear from the context.*

▶ **Example 27** (continued from Example 25)**.** Let $\mathcal{G} = \mathcal{C}_{\mathcal{R}} = \{\mathsf{0}, \mathsf{s}, \mathsf{rand}\}$. For example, the term $t = \mathsf{s}(\mathsf{rand}(\mathsf{0})) -_{\omega+1} \mathsf{s}(\mathsf{0})$ admits the following rewrite sequence of $\mathsf{F}(\mathcal{G})$:

$$\mathsf{s}(\mathsf{rand}(\mathsf{0})) -_{\omega+1} \mathsf{s}(\mathsf{0}) \;\to\; \mathsf{c}_1(\mathsf{rand}(\mathsf{0})) -_{\omega+1} \mathsf{s}(\mathsf{0}) \;\to\; \mathsf{rand}(\mathsf{0}) -_{\omega+1} \mathsf{s}(\mathsf{0}) \;\to^*\; \mathsf{c}_0 -_{\omega+1} \mathsf{c}_0$$

Thus, we obtain $t{\downarrow} = \mathsf{c}_0 -_{\omega+1} \mathsf{c}_0$.

▶ **Example 28.** To see what happens if binary symbols and constant symbols are flattened, let us consider the signature $\mathcal{F} = \{\mathsf{a}^{(0)}, \mathsf{b}^{(0)}, \mathsf{f}^{(2)}\}$ and its subset $\mathcal{G} = \{\mathsf{a}, \mathsf{f}\}$ of $\mathcal{F}$. The terms $\mathsf{f}(\mathsf{f}(\mathsf{a}, x), \mathsf{a})$, $\mathsf{f}(\mathsf{f}(\mathsf{a}, x), \mathsf{b})$, $\mathsf{f}(\mathsf{f}(\mathsf{b}, x), \mathsf{b})$ are flattened into x, $\mathsf{c}_2(x, \mathsf{b})$, $\mathsf{c}_3(\mathsf{b}, x, \mathsf{b})$, respectively.

Since flattening introduces compound symbols in $\mathcal{F}_{\mathsf{c}}$, we extend the $\mathcal{A}$-induced precedence pair $(\succsim, \succ)$ on $\mathcal{F}_{\mathsf{lab}}$ by adjoining all compound symbols as minimal elements. To be precise, the extended precedence pair $(\succsim', >')$ is given by the following conditions:

- $f \succsim' g$ if $f \succsim g$ or $g \in \mathcal{F}_{\mathsf{c}}$.
- $f \succ' g$ if $f \succ g$, or $f \in \mathcal{F}_{\mathsf{lab}}$ and $g \in \mathcal{F}_{\mathsf{c}}$.

Obviously, the pair $(\succsim', \succ')$ is a precedence pair satisfying $\succsim \subseteq \succsim'$ and $\succ \subseteq \succ'$, and well-founded if $(\succsim, \succ)$ is so.

The key observation is that, any rewrite sequence of $\mathcal{R}_{\mathsf{lab}}/\mathcal{S}_{\mathsf{lab}}$ gives rise to a corresponding rewrite sequence of $(\succsim'_{\mathsf{mpo}}, \succ'_{\mathsf{mpo}})$.

▶ **Example 29** (again continued from Example 25)**.** Consider the rewrite sequence:

$$\mathsf{s}(\mathsf{rand}(\mathsf{0})) -_{\omega+1} \mathsf{s}(\mathsf{0}) \;\to_{\mathcal{R}_{\mathsf{lab}}}\; \mathsf{rand}(\mathsf{0}) -_{\omega} \mathsf{0} \;\to_{\mathcal{S}_{\mathsf{lab}}}\; \mathsf{rand}(\mathsf{s}(\mathsf{0})) -_{\omega} \mathsf{0}$$

Let $\mathcal{G} = \mathcal{C}_{\mathcal{R}} = \{\mathsf{rand}, \mathsf{s}, \mathsf{0}\}$. Flattening turns the rewrite sequence into the descending sequence of MPO, namely $\mathsf{c}_0 -_{\omega+1} \mathsf{c}_0 \succ'_{\mathsf{mpo}} \mathsf{c}_0 -_{\omega} \mathsf{c}_0 \succsim'_{\mathsf{mpo}} \mathsf{c}_0 -_{\omega} \mathsf{c}_0$.

For formally discussing the correspondence above, we introduce a relative termination criterion akin to Proposition 19 in Section 4. This criterion exploits the fact that $s \succ_{\mathcal{G}} t$ implies $s{\downarrow_{\mathcal{G}}} \succ'_{\mathsf{mpo}} t{\downarrow_{\mathcal{G}}}$ when $\mathcal{R} \subseteq \succ_{\mathcal{G}}$. The proof is discussed in the next section.

▶ **Theorem 30.** *Let $\mathcal{R}$ and $\mathcal{S}$ be TRSs over a signature $\mathcal{F}$. Then $\mathcal{R}/\mathcal{S}$ is terminating if there exist a well-founded precedence pair $(\succ, \succsim)$ and a subset $\mathcal{G}$ of $\mathcal{F}$ such that $\mathcal{R} \subseteq \succ_{\mathcal{G}}$, $\mathcal{S}$ is non-duplicating, and we have $\ell \succsim_{\mathcal{G}} r$ or $r \in \mathcal{T}(\mathcal{G}, \mathcal{V})$ for all $\ell \to r \in \mathcal{S}$.*

We arrive at the simulation result for the relative version of the dependency pair method.

▶ **Theorem 31.** *Let $\mathcal{R}$ be a TRS and $(\mathcal{A}, \gtrsim, >)$ a weakly monotone well-founded algebra with $\mathcal{R} \cup \mathcal{S} \subseteq \gtrsim_{\mathcal{A}}$. The following statements hold for $\mathcal{G} = \mathcal{C}_{\mathcal{R}}$.*
1. *The $\mathcal{A}$-induced labeling is a weakly monotone labeling for $(\mathcal{A}, \gtrsim)$.*
2. *The $\mathcal{A}$-induced precedence pair $(\succsim, \succ)$ is well-founded.*
3. *If $\mathsf{DP}(\mathcal{R}) \subseteq >_{\mathcal{A}}$ then $\mathcal{R}_{\mathsf{lab}} \subseteq \succ_{\mathcal{G}}$.*
4. *If $\mathcal{S}$ is non-duplicating then so is $\mathcal{S}_{\mathsf{lab}}$.*
5. *If $\mathcal{R}$ dominates $\mathcal{S}$ then $\mathcal{R}_{\mathsf{lab}}$ also dominates $\mathcal{S}_{\mathsf{lab}}$.*
6. *$\mathcal{D}\mathsf{ec}(\gtrsim) \subseteq \succsim_{\mathcal{G}}$.*

Proof. Analogous to the proof of Theorem 21. ◀

Theorem 31 is indeed an analog of Theorem 21. Suppose that relative termination is shown by the dependency pair method with a reduction pair (Theorem 23). Relative termination of the labeled systems resulting from semantic labeling is shown by the precedence-based termination criterion (Theorem 30). The employed labeling and precedence pair are constructible from the reduction pair (Definition 15).

6 Multiset Path Orders with Flattening

This section is devoted to proving Theorem 30, which is obtained as a corollary of two key theorems. Let $(\succsim, \succ)$ be a well-founded precedence pair on $\mathcal{F}$, $(\succsim', \succ')$ the extended precedence pair (introduced in Section 5), and $\mathcal{G}$ a set of function symbols that are flattened. Hereafter, we consider the signature $\mathcal{F} \cup \mathcal{F}_{\mathsf{c}}$ until the end of the section. For example, substitutions are those of terms over $\mathcal{F} \cup \mathcal{F}_{\mathsf{c}}$. Moreover, for brevity we omit the prime symbol $'$ from $\succsim'$ and $\succ'$.

The first key theorem states that $s \to_{\mathcal{R}} t$ implies $s{\downarrow} \succ_{\mathsf{mpo}} t{\downarrow}$, when $\mathcal{R} \subseteq \succ_{\mathcal{G}}$. To this end, we show that the relation $\succ_{\mathcal{G}}$ is closed under substitutions and flattening. The point is that, in contrast to MPOs, Definition 17 demands a greater term to be headed by a function symbol that is *not* flattened. For example, when $\sigma = \{x \mapsto \mathsf{a}\}$ and $\mathsf{f}^{(2)}, \mathsf{a}^{(0)} \in \mathcal{G}$, it holds that $\mathsf{f}(x, y) >_{\mathsf{mpo}} y$ by any MPO $>_{\mathsf{mpo}}$ but $\mathsf{f}(x, y)\sigma{\downarrow} = y = y\sigma{\downarrow}$.

▶ **Lemma 32.** *If $s \succ_{\mathcal{G}} t$ then $s\sigma \succ_{\mathcal{G}} t\sigma$ for all substitutions σ.*

Proof Sketch. Show $s\sigma \succ_{\mathsf{mpo}} t\sigma$ by induction on the derivation of $s \succ_{\mathcal{G}} t$. ◀

▶ **Lemma 33.** *If $s \succ_{\mathcal{G}} t$ then $s{\downarrow} \succ_{\mathsf{mpo}} t{\downarrow}$.*

The proof of Lemma 33 is in Appendix B.

The implication $s{\downarrow} \succ_{\mathcal{G}} t{\downarrow} \implies C[s]{\downarrow} \succ_{\mathcal{G}} C[t]{\downarrow}$ does not hold in general, as witnessed by $\mathsf{f}(x) \succ_{\mathcal{G}} x$ but $\mathsf{g}(\mathsf{f}(x)) \not\succ_{\mathcal{G}} \mathsf{g}(x)$ for $\mathcal{G} = \varnothing$. However, its super-relation $\succ_{\mathsf{mpo}}$ satisfies the corresponding property $s{\downarrow} \succ_{\mathsf{mpo}} t{\downarrow} \implies C[s]{\downarrow} \succ_{\mathsf{mpo}} C[t]{\downarrow}$. Note that $\succ_{\mathsf{mpo}}$ is closed under contexts but not flattening (consider $\mathsf{c}_1(x) \succ_{\mathsf{mpo}} x$). We prepare auxiliary lemmata.

▶ **Lemma 34.** *The inequality $t \succsim_{\mathsf{mpo}} t{\downarrow}$ holds for all terms t.*

Proof. It follows immediately from the fact that $\succsim_{\mathsf{mpo}}$ is a rewrite relation (Theorem 5) and $\ell \succsim_{\mathsf{mpo}} r$ holds for all $\ell \to r \in \mathsf{F}(\mathcal{G})$. ◀

We say that a term t is *root-rigid* if t is a variable or $t = f(t_1, \dots, t_n)$ with $f \in \mathcal{F} \setminus \mathcal{G}$. It is easy to see that if $t{\downarrow} = c_n(t_1, \dots, t_n)$ then $t_1, \dots, t_n$ are root-rigid.

▶ **Lemma 35.** *Let $C = f(u_1, \dots, u_{i-1}, \square, u_{i+1}, \dots, u_n)$. If $s{\downarrow} \succ_{\mathsf{mpo}} t{\downarrow}$ then $C[s]{\downarrow} \succ_{\mathsf{mpo}} C[t]{\downarrow}$.*

Proof. Suppose $s{\downarrow} \succ_{\mathsf{mpo}} t{\downarrow}$. We show $C[s]{\downarrow} \succ_{\mathsf{mpo}} C[t]{\downarrow}$ by well-founded induction on $C[s]$ with respect to $\to^{+}_{\mathsf{F}(\mathcal{G})}$. If $f \in \mathcal{F} \setminus \mathcal{G}$ then the claim immediately follows by MPO (3). So assume $f \in \mathcal{G} \cup \mathcal{F}_{\mathsf{c}}$. The case when $C{\downarrow} = \square$ is trivial. If C is $\mathsf{F}(\mathcal{G})$-reducible, we can construct a context C' such that $C[s] \to^{+}_{\mathsf{F}(\mathcal{G})} C'[s]$, and the induction hypothesis yields the desired inequality $C[s]{\downarrow} = C'[s]{\downarrow} \succ_{\mathsf{mpo}} C'[t]{\downarrow} = C[t]{\downarrow}$. Otherwise, C is already a flattened context, namely $C = \mathsf{c}_n(u_1, \dots, \square, \dots, u_n)$ with $n \geqslant 2$ and $u_1, \dots, u_n$ root-rigid. An easy case is when $C[s]{\downarrow} = C[s{\downarrow}]$, which can be easily handled by Lemma 34 and closure under contexts of $\succ_{\mathsf{mpo}}$. The remaining case is when $s{\downarrow} = \mathsf{c}_m(s_1, \dots, s_m)$. Since c_0 is minimal with respect to $\succ_{\mathsf{mpo}}$ and $\mathsf{c}_1(s_1)$ is not flattened, we have $m \geqslant 2$, which leads to $m + n - 1 \geqslant 2$. Since $s_1, \dots, s_m$ are root-rigid, $C[s]{\downarrow} = \mathsf{c}_{m+n-1}(u_1, \dots, u_{i-1}, s_1, \dots, s_m, u_{i+1}, \dots, u_n)$ is obtained. We further distinguish three cases.

1. If $s{\downarrow} \succ_{\mathsf{mpo}} t{\downarrow}$ is derived by MPO (1), $s_i \succsim_{\mathsf{mpo}} t{\downarrow}$ holds for some $1 \leqslant i \leqslant m$. Because $m \geqslant 2$, we have $\{s_1, \dots, s_m\} \succ^{\mathsf{mul}}_{\mathsf{mpo}} \{t{\downarrow}\}$. So we obtain

   $$\{u_1, \dots, u_{i-1}, s_1, \dots, s_m, u_{i+1}, \dots, u_n\} \succ^{\mathsf{mul}}_{\mathsf{mpo}} \{u_1, \dots, u_{i-1}, t{\downarrow}, u_{i+1}, \dots, u_n\}$$

   from which $C[s]{\downarrow} \succ_{\mathsf{mpo}} C[t{\downarrow}]$ follows by MPO (3).[1] Because $C[t{\downarrow}] \succsim_{\mathsf{mpo}} C[t]{\downarrow}$, the compatibility of $\succsim_{\mathsf{mpo}}$ and $\succ_{\mathsf{mpo}}$ entails the claim.
2. If $s{\downarrow} \succ_{\mathsf{mpo}} t{\downarrow}$ is derived by MPO (2), the root symbol of t is smaller than c_m. This contradicts the minimality of c_m.
3. If $s{\downarrow} \succ_{\mathsf{mpo}} t{\downarrow}$ is derived by MPO (3), $t{\downarrow}$ is of the form $\mathsf{c}_k(t_1, \dots, t_k)$ and the inequality $\{s_1, \dots, s_m\} \succ_{\mathsf{mpo}} \{t_1, \dots, t_k\}$ holds. So $C[s]{\downarrow} = \mathsf{c}_{n+m-1}(u_1, \dots, s_1, \dots, s_m, \dots, u_n)$ and $C[t]{\downarrow} = \mathsf{c}_{n+k-1}(u_1, \dots, t_1, \dots, t_k, \dots, u_n)$. Thus, $C[s]{\downarrow} \succ_{\mathsf{mpo}} C[t]{\downarrow}$ follows by MPO (3). ◀

▶ **Lemma 36.** *If $s{\downarrow} \succ_{\mathsf{mpo}} t{\downarrow}$ then $C[s]{\downarrow} \succ_{\mathsf{mpo}} C[t]{\downarrow}$ for all contexts C.*

Proof. The claim is shown by straightforward structural induction on C using Lemma 35. ◀

Combining these properties, we obtain the first key theorem.

▶ **Theorem 37.** *Let $\mathcal{R}$ be a TRS with $\mathcal{R} \subseteq \succ_{\mathcal{G}}$ and let s and t be terms. If $s \to_{\mathcal{R}} t$ then $s{\downarrow} \succ_{\mathsf{mpo}} t{\downarrow}$.*

Proof. Let $s \to_{\mathcal{R}} t$. There exist a rule $\ell \to r \in \mathcal{R}$, a substitution σ, and a context C such that $s = C[\ell\sigma]$ and $t = C[r\sigma]$. Since $\ell \succ_{\mathcal{G}} r$ holds by assumption, we obtain the implications:

$$\ell \succ_{\mathcal{G}} r \overset{32}{\Longrightarrow} \ell\sigma \succ_{\mathcal{G}} r\sigma \overset{33}{\Longrightarrow} \ell\sigma{\downarrow} \succ_{\mathsf{mpo}} r\sigma{\downarrow} \overset{36}{\Longrightarrow} C[\ell\sigma]{\downarrow} \succ_{\mathsf{mpo}} C[r\sigma]{\downarrow}$$

Here the numbers indicate the employed lemmata. Thus, $s{\downarrow} \succ_{\mathsf{mpo}} t{\downarrow}$ holds. ◀

The second key theorem states that $s \to_{\mathcal{S}} t$ implies $s{\downarrow} \succsim_{\mathsf{mpo}} t{\downarrow}$, provided that $\mathcal{S}$ is non-duplicating and $\ell \succsim_{\mathcal{G}} r$ or $r \in \mathcal{T}(\mathcal{G}, \mathcal{V})$ holds for all $\ell \to r \in \mathcal{S}$. To this end we show that, if $\ell \to r \in \mathcal{S}$ then $\ell\sigma{\downarrow} \succsim_{\mathsf{mpo}} r\sigma{\downarrow}$. The next lemma addresses the case when $s \to_{\mathcal{S}} t$ employs a rule $\ell \to r$ with $\ell \succsim_{\mathcal{G}} r$.

---

[1] For example, consider $C = \mathsf{c}_2(\square, y)$, $s = \mathsf{c}_2(x, x)$, and $t = x$. The inequality $s{\downarrow} = \mathsf{c}_2(x, x) \succ_{\mathsf{mpo}} x = t{\downarrow}$ is derived by MPO (1), while $C[s]{\downarrow} = \mathsf{c}_3(x, x, y) \succ_{\mathsf{mpo}} \mathsf{c}_2(x, y) = C[t]{\downarrow}$ is derived by MPO (3).

▶ **Lemma 38.** *If $\ell \gtrsim_{\mathcal{G}} r$ then $\ell\sigma{\downarrow} \gtrsim_{\mathsf{mpo}} r\sigma{\downarrow}$ for all substitutions σ.*

Proof. The proof is analogous to Lemma 32 and Lemma 33. ◀

The other case when $s \to_{\mathcal{S}} t$ uses a rule $\ell \to r$ with $r \in \mathcal{T}(\mathcal{G}, \mathcal{V})$ is more difficult. Actually $s{\downarrow} \gtrsim_{\mathsf{mpo}} t{\downarrow}$ does not hold in general. Consider the case that $s = \ell = \mathsf{f}(x, \mathsf{a})$ and $t = r = \mathsf{a}$ with $\mathsf{f}, \mathsf{a} \in \mathcal{G}$. Because $s{\downarrow} = x$ and $t{\downarrow} = \mathsf{c}_0$, the inequality $s{\downarrow} \gtrsim_{\mathsf{mpo}} t{\downarrow}$ does not hold. Fortunately, the claim holds when s is ground.

▶ **Lemma 39.** *If s is ground and $s \trianglerighteq t$ then $s{\downarrow} \gtrsim_{\mathsf{mpo}} t{\downarrow}$.*

The proof is in Appendix B. It is essential for Lemma 39 that s is ground, as seen by the example above.

▶ **Lemma 40.** *Let $t = f(t_1, \ldots, t_n)$. If $\mathcal{F}\mathsf{un}(t) \cap \mathcal{G} = \varnothing$ and $t \in \mathsf{NF}(\xrightarrow{\epsilon}_{\mathsf{F}(\mathcal{G})})$ then $t{\downarrow} = f(t_1{\downarrow}, \ldots, t_n{\downarrow})$.* ◀

▶ **Lemma 41.** *Let (s, t) be a non-duplicating pair with $t \in \mathcal{T}(\mathcal{G} \cup \mathcal{F}_{\mathsf{c}}, \mathcal{V})$ and σ a grounding substitution for s and t. The relation $s\sigma{\downarrow} \gtrsim_{\mathsf{mpo}} t\sigma{\downarrow}$ holds.*

Proof. We show the claim by induction on $s\sigma$ with respect to $\to^{+}_{\mathsf{F}(\mathcal{G})}$. The case when $t{\downarrow}$ is a variable is routine. So suppose $t{\downarrow} = \mathsf{c}_n(x_1, \ldots, x_n)$. Depending on reducibility of $s\sigma$ by $\mathsf{F}(\mathcal{G})$, we distinguish several cases.

1. If s is not flattened then $s\sigma \to^{+}_{\mathsf{F}(\mathcal{G})} s{\downarrow}\sigma$. Since $|s{\downarrow}|_x = |s|_x \geqslant |t|_x$ for all variables x, the induction hypothesis yields $s\sigma{\downarrow} = (s{\downarrow}\sigma){\downarrow} \gtrsim_{\mathsf{mpo}} t\sigma{\downarrow}$.
2. If $x\sigma$ is not flattened for some $x \in \mathcal{V}\mathsf{ar}(s)$ then $s\sigma \to^{+}_{\mathsf{F}(\mathcal{G})} s\tau$, where τ is given by $\tau(y) = y\sigma{\downarrow}$ for each variable y. By the induction hypothesis $s\sigma{\downarrow} = s\tau{\downarrow} \gtrsim_{\mathsf{mpo}} t\tau{\downarrow} = t\sigma{\downarrow}$ follows.
3. If the last two conditions are not satisfied and $s\sigma \notin \mathsf{NF}(\xrightarrow{\epsilon}_{\mathsf{F}(\mathcal{G})})$ then $s = \mathsf{c}_m(s_1, \ldots, s_m)$ with s_i a variable and $s_i\sigma{\downarrow} = \mathsf{c}_k(u_1, \ldots, u_k)$. Recall $t{\downarrow} = \mathsf{c}_n(x_1, \ldots, x_n)$.
   - If $s_i = x_j$ for some $1 \leqslant j \leqslant n$ then consider the terms
$$s' = \mathsf{c}_{m+k-1}(s_1, \ldots, s_{i-1}, y_1, \ldots, y_k, s_{i+1}, \ldots, s_m)$$
$$t' = \mathsf{c}_{n+k-1}(x_1, \ldots, x_{j-1}, y_1, \ldots, y_k, x_{j+1}, \ldots, x_n)$$
     and the substitution τ given by $\tau(y_h) = u_h$ for $h \in \{1, \ldots, k\}$ and $\tau(z) = \sigma(z)$ for other variables z, where $y_1, \ldots, y_k$ are fresh variables. Since $s\sigma \to^{+}_{\mathsf{F}(\mathcal{G})} s'\tau$ and $|s'|_x \geqslant |t'|_x$ for all variables x, by the induction hypothesis we obtain $s\sigma{\downarrow} = s'\tau{\downarrow} \gtrsim_{\mathsf{mpo}} t'\tau{\downarrow} = t\sigma{\downarrow}$.
   - Otherwise, consider the term $s' = \mathsf{c}_{m+k-1}(s_1, \ldots, s_{i-1}, u_1, \ldots, u_k, s_{i+1}, \ldots, s_m)$. We have $s\sigma \to^{+}_{\mathsf{F}(\mathcal{G})} s'\sigma$ and $|s'|_x \geqslant |t|_x$ for all variables x. By the induction hypothesis $s\sigma{\downarrow} = s'\sigma{\downarrow} \gtrsim_{\mathsf{mpo}} t\sigma{\downarrow}$ follows.
4. Otherwise, s and $x\sigma$ for all $x \in \mathcal{V}\mathsf{ar}(s)$ are flattened, and $s\sigma \in \mathsf{NF}(\xrightarrow{\epsilon}_{\mathsf{F}(\mathcal{G})})$. Since the former condition guarantees $\mathcal{F}\mathsf{un}(s\sigma) \cap \mathcal{G} = \varnothing$, Lemma 40 yields $s\sigma{\downarrow} = f(s_1\sigma{\downarrow}, \ldots, s_m\sigma{\downarrow})$. Recalling $t{\downarrow} = \mathsf{c}_n(x_1, \ldots, x_n)$, we further distinguish two cases.
   (a) If $f \in \mathcal{F} \setminus \mathcal{G}$ then $f \succ \mathsf{c}_n$. In addition, $s{\downarrow} \succ_{\mathcal{G}} x_j$ holds for all j. Since $\succ_{\mathcal{G}}$ is closed under substitutions and flattening (Lemmata 32 and 33), $s\sigma{\downarrow} \succ_{\mathcal{G}} x_j\sigma{\downarrow}$ is obtained. Thus, $s\sigma{\downarrow} \succ_{\mathsf{mpo}} x_j\sigma{\downarrow}$. Hence, MPO (2) entails the claim.
   (b) Otherwise, $f = \mathsf{c}_m$ and $s_i\sigma$ are root-rigid for all i. Let Y_i denote the multiset of variables in s_i. Since $|s|_x \geqslant |t|_x$ for all variables x, the multiset $\{x_1, \ldots, x_n\}$ can be represented by $X_1 \uplus \cdots \uplus X_m$ with $X_i \subseteq Y_i$ for each $1 \leqslant i \leqslant m$. First we show the subgoal $\{s_i\sigma{\downarrow}\} \gtrsim^{\mathsf{mul}}_{\mathsf{mpo}} \{x\sigma{\downarrow} \mid x \in X_i\}$:
   - If $X_i = \varnothing$ then the claim is trivial.

- If $X_i = \{x\}$ for some variable x then $s_i \trianglerighteq x$, and thus $s_i\sigma \trianglerighteq x\sigma$. By Lemma 39 the claim follows.
- Otherwise, $|Y_i| \geqslant |X_i| \geqslant 2$. As s_i contains at least two variables, s_i is not a variable. So for every $x \in X_i$ we have $s_i \vartriangleright x$. As $\mathsf{root}(s_i) \in \mathcal{F} \setminus \mathcal{G}$, the relation $s_i \succ_{\mathcal{G}} x$ holds. As in Case (a), we can deduce $s_i\sigma{\downarrow} \succ_{\mathsf{mpo}} x\sigma{\downarrow}$. Thus, the claim holds.

The subgoal results in $\{s_1\sigma{\downarrow}, \dots, s_m\sigma{\downarrow}\} \succsim^{\mathsf{mul}} \{x_1\sigma{\downarrow}, \dots, x_n\sigma{\downarrow}\}$. Thus, the inequalities

$$s\sigma{\downarrow} = \mathsf{c}_m(s_1\sigma{\downarrow}, \dots, s_m\sigma{\downarrow}) \succsim_{\mathsf{mpo}} \mathsf{c}_n(x_1\sigma{\downarrow}, \dots, x_n\sigma{\downarrow}) \succsim_{\mathsf{mpo}} t\sigma{\downarrow}$$

are obtained by MPO (3) and Lemma 34. ◀

As in the case of $\succ_{\mathsf{mpo}}$ (Lemmata 35 and 36), one can verify that $\succsim_{\mathsf{mpo}}$ is preserved under the combination of context application and flattening.

▶ **Lemma 42.** *If $s{\downarrow} \succsim_{\mathsf{mpo}} t{\downarrow}$ then $C[s]{\downarrow} \succsim_{\mathsf{mpo}} C[t]{\downarrow}$ for all contexts C.* ◀

We arrive at the second key theorem for $\mathcal{S}$-steps.

▶ **Theorem 43.** *Let $\mathcal{S}$ be a non-duplicating TRS such that $\ell \succsim_{\mathcal{G}} r$ or $r \in \mathcal{T}(\mathcal{G}, \mathcal{V})$ for all $\ell \to r \in \mathcal{S}$. Let s and t be ground terms. If $s \to_{\mathcal{S}} t$ then $s{\downarrow} \succsim_{\mathsf{mpo}} t{\downarrow}$.*

Proof. Let $s \to_{\mathcal{S}} r$. There exist a rule $\ell \to r \in \mathcal{S}$, a grounding substitution σ for s and t, and a context C such that $s = C[\ell\sigma]$ and $t = C[r\sigma]$. We have the following implications:

$$\ell \to r \in \mathcal{S} \implies \ell\sigma{\downarrow} \succsim_{\mathsf{mpo}} r\sigma{\downarrow} \overset{42}{\implies} C[\ell\sigma]{\downarrow} \succ_{\mathsf{mpo}} C[r\sigma]{\downarrow}$$

The first implication follows from Lemma 38 or Lemma 41. Thus, $s{\downarrow} \succsim_{\mathsf{mpo}} t{\downarrow}$ holds. ◀

Theorem 30 is a consequence of Theorems 37 and 43.

Proof of Theorem 30. It suffices to show termination of $\mathcal{R}/\mathcal{S}$ under the extended signature $\mathcal{F} \cup \mathcal{F}_{\mathsf{c}}$. Since $\mathcal{F}_{\mathsf{c}}$ contains the constant c_0, every infinite rewrite sequence of terms can turn into an infinite rewrite sequence of ground terms by instantiating variables to c_0. Therefore, our task boils down to proving termination on ground terms. Consider ground terms s and t. By Theorems 37 and 43 we obtain the implications:

$$s \to_{\mathcal{R}/\mathcal{S}} t \implies s \to_{\mathcal{S}}^{*} \cdot \to_{\mathcal{R}} \cdot \to_{\mathcal{S}}^{*} t \implies s{\downarrow} \succsim_{\mathsf{mpo}} \cdot \succ_{\mathsf{mpo}} \cdot \succsim_{\mathsf{mpo}} t{\downarrow} \implies s{\downarrow} \succ_{\mathsf{mpo}} t{\downarrow}$$

As $\succ_{\mathsf{mpo}}$ is well-founded, $\to_{\mathcal{R}/\mathcal{S}}$ is terminating on ground terms. ◀

7 Improving Applicability

Theorem 23 is a simple and elegant adaptation of the original method (Theorem 11). However, the dominance condition can be a severe restriction, for example, in confluence analysis based on relative termination.

▶ **Example 44.** Consider the confluence problem of the following TRS:

1:	$x + \mathsf{s}(y) \to \mathsf{s}(x) + y$	6:	$x \times \mathsf{s}(y) \to x + (x \times y)$
2:	$\mathsf{s}(x) + y \to x + \mathsf{s}(y)$	7:	$\mathsf{s}(x) \times y \to (x \times y) + y$
3:	$x + y \to y + x$	8:	$x \times y \to y \times x$
4:	$(x + y) + z \to x + (y + z)$	9:	$\mathsf{sq}(x) \to x \times x$
5:	$x + (y + z) \to (x + y) + z$	10:	$\mathsf{sq}(\mathsf{s}(x)) \to (x \times x) + \mathsf{s}(x + x)$

By the rule labeling technique by Zankl et al. (see [28, Example 17]) the confluence problem boils down to the relative termination problem of $\mathcal{R}/\mathcal{S}$, where $\mathcal{R}$ is the set of all duplicating rules in the above TRS and $\mathcal{S}$ is the set of the non-duplicating rules; namely $\mathcal{R} = \{6,7,9,10\}$ and $\mathcal{S} = \{1,2,3,4,5,8\}$. In order to use Theorem 23 it is necessary that $\mathcal{S}$ is non-duplicating and $\mathcal{R}$ dominates $\mathcal{S}$. The former is satisfied, but the latter does not hold. In fact, $\mathcal{D}_\mathcal{R} = \{\times, \mathsf{sq}\}$ and $\times$ occurs in the right-hand side of rule 8. So Theorem 23 is not applicable to this example.

Multiset path orders are capable of dealing with termination modulo commutative (or permutative) axioms, where Theorem 23 fails due to absence of dominance. Using our simulation technique, we incorporate this advantage into the dependency pair method for relative termination.

We introduce a generalized notion of dominance. We say that a rule is *transitional* if it is of the form $f(x_1,\dots,x_n) \to f(y_1,\dots,y_n)$ with $x_1,\dots,x_n,y_1,\dots,y_n$ variables. Note that the variables need not be different from each other.

▶ **Definition 45.** *Let $\mathcal{R}$ and $\mathcal{S}$ be TRSs over the same signature and $\mathcal{G} = \mathcal{C}_\mathcal{R}$. We say that $\mathcal{R}$* almost dominates *$\mathcal{S}$ if for every $\ell \to r \in \mathcal{S}$ either $r \in \mathcal{T}(\mathcal{C}_\mathcal{R},\mathcal{V})$ or $\ell \to r{\downarrow_\mathcal{G}}$ is a transitional rule with $\mathsf{root}(\ell) \in \mathcal{D}_\mathcal{R}$.*

We incorporate $\mathsf{DP}_{\mathcal{D}_\mathcal{R}}(\mathcal{S})$ (see Definition 10) into Theorem 23. We denote this set by $\mathsf{DP}(\mathcal{S})$. When a rule $\ell \to r \in \mathcal{S}$ satisfies the second condition in Definition 45, it gives rise to exactly one dependency pair.

▶ **Theorem 46.** *Suppose that a TRS $\mathcal{R}$ almost dominates a non-duplicating TRS $\mathcal{S}$. Then $\mathcal{R}/\mathcal{S}$ is terminating if there is a reduction pair $(\gtrsim, >)$ with $\mathsf{DP}(\mathcal{R}) \subseteq {>}$ and $\mathcal{R} \cup \mathcal{S} \cup \mathsf{DP}(\mathcal{S}) \subseteq {\gtrsim}$.*

▶ Remark 47. As discussed in Remark 12, if Theorem 46 is applicable, relative termination follows from finiteness of $(\mathsf{DP}(\mathcal{R}), \mathcal{R} \cup \mathcal{S} \cup \mathsf{DP}(\mathcal{S}))$. So all methods for the dependency pair framework [13], including the iterative use of reduction pairs [14] and dependency graph techniques [4], are available for showing relative termination.

For proving the theorem we need an improved version of Theorem 30.

▶ **Theorem 48.** *Let $\mathcal{R}$ and $\mathcal{S}$ be TRSs over a signature $\mathcal{F}$. Then $\mathcal{R}/\mathcal{S}$ is terminating if there exist a well-founded precedence pair $(\succsim, \succ)$ and a subset $\mathcal{G}$ of $\mathcal{F}$ such that $\mathcal{R} \subseteq {\succ_\mathcal{G}}$, $\mathcal{S}$ is non-duplicating, and every $\ell \to r \in \mathcal{S}$ satisfies one of the following alternatives:*

1. $\ell \succsim_\mathcal{G} r$
2. $r \in \mathcal{T}(\mathcal{G},\mathcal{V})$
3. $\ell = f(x_1,\dots,x_m) \succsim_{\mathsf{mpo}} g(y_1,\dots,y_n) = r{\downarrow_\mathcal{G}}$ *for some variables $x_1,\dots,x_m,y_1,\dots,y_n$ and $f, g \in \mathcal{F} \setminus \mathcal{G}$*

Here $\succ_\mathcal{G}$, $\succsim_\mathcal{G}$ and $\succsim_{\mathsf{mpo}}$ are the relations induced by the precedence pair $(\succsim, \succ)$.

Proof. The proof is analogous to that of Theorem 30, but we need to extend Theorem 43 to cover the case when $\ell \to r \in \mathcal{S}$ satisfies the third condition. This is archived by showing $\ell\sigma{\downarrow} \succsim_{\mathsf{mpo}} r\sigma{\downarrow}$ for all substitutions σ. The inequality is verified by easy case distinction. ◀

Proof of Theorem 46. The proof follows the simulation result of Section 5. Given a reduction pair $(\geqslant_\mathcal{A}, >_\mathcal{A})$, we label $\mathcal{R}$ and $\mathcal{S}$ with the $\mathcal{A}$-induced labeling. By taking the $\mathcal{A}$-induced precedence and $\mathcal{G} = \mathcal{C}_\mathcal{R}$ termination of $\mathcal{R}_{\mathsf{lab}}/(\mathcal{S}_{\mathsf{lab}} \cup \mathcal{D}\mathsf{ec}(\geqslant_\mathcal{A}))$ follows from Theorem 48. Hence the claim holds. Note that $\mathsf{DP}(\mathcal{S}) \subseteq {\geqslant_\mathcal{A}}$ guarantees $f_a \succsim g_b$ in the case of transitional rules in Definition 45, and therefore handled by the third case of Theorem 48. ◀

▶ **Example 49** (continued from Example 44). Now we switch from Theorem 23 to Theorem 46. Recalling $\mathcal{D}_{\mathcal{R}} = \{\times, \mathsf{sq}\}$, we can easily see that $\mathcal{R}$ almost dominates $\mathcal{S}$. The set $\mathsf{DP}(\mathcal{R})$ of dependency pairs consists of the four rules

$$x \times^\sharp \mathsf{s}(y) \to x \times^\sharp y \qquad \mathsf{s}(x) \times^\sharp y \to x \times^\sharp y \qquad \mathsf{sq}^\sharp(x) \to x \times^\sharp x \qquad \mathsf{sq}^\sharp(\mathsf{s}(x)) \to x \times^\sharp x$$

and $\mathsf{DP}(\mathcal{S}) = \{x \times^\sharp y \to y \times^\sharp x\}$. Take the following weakly monotone algebra $\mathcal{A}$ on $\mathbb{N}$:

$$\mathsf{s}_{\mathcal{A}}(x) = x + 1 \quad x +_{\mathcal{A}} y = 0 \quad x \times_{\mathcal{A}} y = x \times^\sharp_{\mathcal{A}} y = x + y \quad \mathsf{sq}_{\mathcal{A}}(x) = \mathsf{sq}^\sharp_{\mathcal{A}}(x) = 2x + 1$$

The reduction pair $(\geqslant_{\mathcal{A}}, >_{\mathcal{A}})$ satisfies $\mathsf{DP}(\mathcal{R}) \subseteq >_{\mathcal{A}}$ and $\mathcal{R} \cup \mathcal{S} \cup \mathsf{DP}(\mathcal{S}) \subseteq \geqslant_{\mathcal{A}}$, and therefore $\mathcal{R}/\mathcal{S}$ is terminating. Observe that the proof via Theorem 46 only uses linear polynomials, but a termination proof of $\mathcal{R}/\mathcal{S}$ by polynomial interpretation over $\mathbb{N}$ demands quadratic ones.

8 Conclusion

We conclude the paper by discussing experimental results and related work.

Evaluation by experiments. In order to assess practicality we have implemented a prototype tool for relative termination based on the improved dependency pair method (Theorem 46).[2] Following Remark 47, the tool attempts to prove finiteness of the corresponding dependency pair problem by iterative application of reduction pairs based on ordinal interpretations below ω^3 as in Example 24. The tool shows relative termination of 48 problems in the `TRS_Relative` category of TPDB 11.3 [24], which consists of 98 problems. There are 8 problems that satisfy the relaxed preconditions (non-duplicatingness and almost dominance) of Theorem 46 but not dominance of Theorem 23 due to Iborra et al. Among them, 6 problems are proved terminating. While all the 6 problems are solved by at least one of existing tools, the use of the dependency pair method (Theorem 46) often makes proofs easier. For example, the problem `Relative_05/rt2-1` asks to show the relative termination of $\{\mathsf{T}(\mathsf{I}(x), y) \to \mathsf{T}(x, y)\}$ with respect to $\{\mathsf{T}(x, y) \to \mathsf{T}(x, \mathsf{I}(y))\}$. Since the almost dominance condition holds, the dependency pair method with the linear polynomial interpretation $\mathsf{T}_{\mathcal{A}}(x, y) = \mathsf{T}^\sharp_{\mathcal{A}}(x, y) = x$ and $\mathsf{I}_{\mathcal{A}}(x) = x + 1$ proves the relative termination. In contrast, the 2023 version of AProVE and $\mathsf{T_TT_2}$ solve this problem by using two- and five-dimensional matrix interpretations, respectively.

Correctness proofs for dependency pairs. Using a model-based version of semantic labeling [29, Theorem 4], Arts showed correctness of an earlier version of the dependency pair method [3, Theorem 9]. In contrast to Theorem 21, this proof is involved and restricted to constructor TRSs. Later, the proof was simplified by switching to a direct proof based on the notion of minimal non-terminating term [4]. This became the standard proof method. In the presented work we re-introduced semantic labeling. A key difference is that our work adopts the one based on quasi-models ([29, Theorem 8] and [12, Corollary 1]).

Potential future work is to extend the presented simulation methodology to dependency pair methods for other rewriting formats. We anticipate that, with suitable semantic labeling, AC-dependency pairs [1, 27] can be simulated by Rubio's AC-RPO [21] and that dependency

[2] The tool and the experimental data (including comparison to existing termination tools) are available at: `https://www.jaist.ac.jp/project/saigawa/24fscd/`.

pairs based on strong computability [18, 10] can be simulated by higher-order RPO [16]. This is not only of theoretical interest, since thus-obtained proofs might ease formalization in proof assistants, provided signatures extension (caused by labeling) can be smoothly handled, see [2, 7, 22] for related discussions.

Completeness of semantic labeling and precedence termination. It is known that semantic labeling with precedence termination (cf. Proposition 19) is a complete method for showing termination of TRSs [19, Theorem 4], meaning that if a TRS is terminating then the termination is shown by semantic labeling and a simpler version of precedence termination. We remark that the combination of Theorems 11 and 21 yields a similar result. As for relative termination, the combination of Theorems 23 and 31 shows completeness of semantic labeling for TRSs with dominance and non-duplicatingness. It is future work to extend this result to a wider class of TRSs.

References

1. B. Alarcón, S. Lucas, and J. Meseguer. A dependency pair framework for A∨C-termination. In *Proceedings of the 8th International Workshop on Rewriting Logic and its Applications*, LNCS, pages 36–52, 2010. doi:10.1007/978-3-642-16310-4_4.
2. A. Almeida and M. Ayala-Rincón. Formalizing the dependency pair criterion for innermost termination. *Science of Computer Programming*, 195:102474, 2020. doi:10.1016/j.scico.2020.102474.
3. T. Arts. Termination by absence of infinite chains of dependency pairs. In *Proc. 21st International Colloquium on Trees in Algebra and Programming*, volume 1059 of *LNCS*, pages 196–210, 1996. doi:10.1007/3-540-61064-2_38.
4. T. Arts and J. Giesl. Termination of term rewriting using dependency pairs. *Theoretical Computer Science*, 236:133–178, 2000. doi:10.1016/S0304-3975(99)00207-8.
5. F. Baader and T. Nipkow. *Term Rewriting and All That.* Cambridge University Press, 1998.
6. L. Bachmair and D. Plaisted. Associative path orderings. In *Proc. 1st International Conference on Rewriting Techniques and Applications*, volume 202 of *LNCS*, pages 241–254, 1985. doi:10.1007/3-540-15976-2_11.
7. F. Blanqui and A. Koprowski. CoLoR: a Coq library on well-founded rewrite relations and its application to the automated verification of termination certificates. *Mathematical Structures in Computer Science*, 21(4):827–859, 2011. doi:10.1017/S0960129511000120.
8. N. Dershowitz. Orderings for term-rewriting systems. *Theoretical Computer Science*, 17:279–301, 1982. doi:10.1016/0304-3975(82)90026-3.
9. N. Dershowitz. Termination by abstraction. In *Proc. 20th International Conference on Logic Programming*, volume 3132 of *LNCS*, pages 1–18, 2004. doi:10.1007/978-3-540-27775-0_1.
10. C. Fuhs and C. Kop. A static higher-order dependency pair framework. In *Proceedings of ESOP 2019*, volume 11423 of *LNCS*, pages 752–782, 2019. doi:10.1007/978-3-030-17184-1_27.
11. A. Geser. *Relative Termination.* PhD thesis, Universität Passau, 1990. URL: http://vts.uni-ulm.de/docs/2012/8146/vts_8146_11884.pdf.
12. A. Geser. Semantic labelling followed by recursive path order with status is strictly more powerful than the corresponding semantic path order. Technical report, Universität Passau, 1994.
13. J. Giesl, R. Thiemann, and P. Schneider-Kamp. The dependency pair framework: Combining techniques for automated termination proofs. In *Proc. 11th International Conference on Logic Programming and Automated Reasoning*, volume 3452 of *LNCS (LNAI)*, pages 301–331, 2005. doi:10.1007/978-3-540-32275-7_21.
14. Nao Hirokawa and Aart Middeldorp. Automating the dependency pair method. *Information and Computation*, 199:172–199, 2005. doi:10.1016/j.ic.2004.10.004.

15 J. Iborra, N. Nishida, G. Vidal, and A. Yamada. Relative termination via dependency pairs. *Journal of Automated Reasoning*, 58:391–411, 2017. doi:10.1007/s10817-016-9373-5.

16 J.-P. Jouannaud and A. Rubio. The higher-order recursive path ordering. In *Proc. 14th Symposium on Logic in Computer Science*, pages 402–411, 1999. doi:10.1109/LICS.1999.782635.

17 S. Kamin and J.J. Lévy. Two generalizations of the recursive path ordering. Technical report, University of Illinois, 1980. Unpublished manuscript.

18 K. Kusakari, Y. Isogai, M. Sakai, and F. Blanqui. Static dependency pair method based on strong computability for higher-order rewrite systems. *IEICE Transactions on Information and Systems*, E92.D(10):2007–2015, 2009. doi:10.1587/transinf.E92.D.2007.

19 A. Middeldorp, H. Ohsaki, and H. Zantema. Transforming termination by self-labelling. In *Proc. 13th International Conference on Automated Deduction*, volume 1104 of *LNCS*, pages 373–387, 1996. doi:10.1007/3-540-61511-3_101.

20 H. Ohsaki, A. Middeldorp, and J. Giesl. Equational termination by semantic labelling. In *Proc. 14th Annual Conference of the European Association for Computer Science Logic*, volume 1862 of *LNCS*, pages 457–471, 2000. doi:10.1007/3-540-44622-2_31.

21 Albert Rubio. A fully syntactic AC-RPO. *Information and Computation*, 178(2):515–533, 2002. doi:10.1006/inco.2002.3158.

22 C. Sternagel and R. Thiemann. Signature extensions preserve termination - an alternative proof via dependency pairs. In *Proc. 19th Annual Conference of the European Association for Computer Science Logic*, volume 6247 of *LNCS*, pages 514–528, 2010.

23 Terese. *Term Rewriting Systems*. Cambridge University Press, 2003.

24 Termination Community. The Termination Problem Database (TPDB). https://github.com/TermCOMP/TPDB. Accessed: 05.12.2023.

25 R. Thiemann, G. Allais, and J. Nagele. On the formalization of termination techniques based on multiset orderings. In *Proc. 23rd International Conference on Rewriting Techniques and Applications*, volume 15 of *LIPIcs*, pages 339–354, 2012. doi:10.4230/LIPIcs.RTA.2012.339.

26 R. Thiemann, J. Schöpf, C. Sternagel, and A. Yamada. Certifying the weighted path order. In *Proc. 5th International Conference on Formal Structures on Computation and Deduction*, volume 167 of *LIPIcs*, pages 4:1–4:20, 2020. doi:10.4230/LIPICS.FSCD.2020.4.

27 A. Yamada, C. Sternagel, R. Thiemann, and K. Kusakari. AC dependency pairs revisited. In *Proc. 25th Annual Conference of the European Association for Computer Science Logic*, volume 62 of *LIPIcs*, pages 8:1–8:16, 2016. doi:10.4230/LIPICS.CSL.2016.8.

28 H. Zankl, B. Felgenhauer, and A. Middeldorp. Labelings for decreasing diagrams. *Journal of Automated Reasoning*, 54(2):101–133, 2015. doi:10.1007/S10817-014-9316-Y.

29 H. Zantema. Termination of term rewriting by semantic labelling. *Fundamenta Informaticae*, 24:89–105, 1995. doi:10.3233/FI-1995-24124.

A Proof of Geser's Semantic Labeling

To make the paper self-contained, we prove Theorem 7 by recasting Geser's original proof [12], see also [23, Section 6.5.4]. Let $(\mathcal{A}, \gtrsim)$ be a weakly monotone algebra and $\mathcal{L}$ a weakly monotone labeling for $(\mathcal{A}, \gtrsim)$. Suppose that TRSs $\mathcal{R}$ and $\mathcal{S}$ satisfy $\mathcal{R} \cup \mathcal{S} \subseteq \gtrsim_{\mathcal{A}}$.

▶ **Lemma 50** ([23, Lemma 6.5.31]). *The identity* $\mathsf{lab}(t\sigma, \alpha) = \mathsf{lab}(t, \beta)\tau$ *holds for all terms* t*, substitutions* σ*, and assignments* α*. Here* β *and* τ *are the assignment and the substitution defined by* $\beta(x) = [\alpha]_{\mathcal{A}}(x\sigma)$ *and* $\tau(x) = \mathsf{lab}(x\sigma, \alpha)$ *for all variables* x*.* ◀

▶ **Lemma 51** ([12, Theorem 2]). *If* $s \to_{\mathcal{R}} t$ *then* $\mathsf{lab}(s, \alpha) \to_{\mathcal{R}_{\mathsf{lab}}/\mathcal{D}\mathsf{ec}(\gtrsim)} \mathsf{lab}(t, \alpha)$ *for all assignments* α*.*

Proof. We show the statement by induction on $|s| + |t|$. Let $s \to_{\mathcal{R}} t$ and let α be an assignment. Depending on the rewrite position of $s \to_{\mathcal{R}} t$, we distinguish two cases.

- If the rewrite position is root then there exist $\ell \to r \in \mathcal{R}$ and a substitution σ such that $s = \ell\sigma$ and $t = r\sigma$. From Lemma 50 we obtain $\mathsf{lab}(s, \alpha) \to_{\mathcal{R}_{\mathsf{lab}}} \mathsf{lab}(t, \alpha)$ as follows:

$$\mathsf{lab}(s, \alpha) = \mathsf{lab}(\ell\sigma, \alpha) = \mathsf{lab}(\ell, \beta)\tau \to_{\mathcal{R}_{\mathsf{lab}}} \mathsf{lab}(r, \beta)\tau = \mathsf{lab}(r\sigma, \alpha) = \mathsf{lab}(t, \alpha)$$

- Suppose that $s = f(s_1, \ldots, s_i, \ldots, s_n)$ and the i-th argument s_i is rewritten. In this case, we have $t = f(s_1, \ldots, t', \ldots, s_n)$ and $s_i \to_{\mathcal{R}} t'$. The induction hypothesis yields $\mathsf{lab}(s_i, \alpha) \to_{\mathcal{R}_{\mathsf{lab}}/\mathcal{D}\mathsf{ec}(\gtrsim)} \mathsf{lab}(t', \alpha)$. Let $a = \mathsf{lab}_f([\alpha]_{\mathcal{A}}(s_1), \ldots, [\alpha]_{\mathcal{A}}(s_i), \ldots, [\alpha]_{\mathcal{A}}(s_n))$ and $b = \mathsf{lab}_f([\alpha]_{\mathcal{A}}(s_1), \ldots, [\alpha]_{\mathcal{A}}(t'), \ldots, [\alpha]_{\mathcal{A}}(s_n))$. From $\mathcal{R} \subseteq {\gtrsim_{\mathcal{A}}}$ and that $\gtrsim_{\mathcal{A}}$ is a rewrite preorder, we have $s_i \gtrsim_{\mathcal{A}} t'$. Moreover, since $\mathcal{L}$ is weakly monotone, the inequality $a \gtrsim b$ holds. So $f_a(x_1, \ldots, x_n) \to f_b(x_1, \ldots, x_n) \in \mathcal{D}\mathsf{ec}(\gtrsim)$. Finally, we obtain the claim as follows:

$$\begin{aligned}
\mathsf{lab}(s, \alpha) ={} & f_a(\mathsf{lab}(s_1, \alpha), \ldots, \mathsf{lab}(s_i, \alpha), \ldots, \mathsf{lab}(s_n, \alpha)) \\
\to_{\mathcal{R}_{\mathsf{lab}}/\mathcal{D}\mathsf{ec}(\gtrsim)} {} & f_a(\mathsf{lab}(s_1, \alpha), \ldots, \mathsf{lab}(t', \alpha) \ldots, \mathsf{lab}(s_n, \alpha)) \\
\to_{\mathcal{D}\mathsf{ec}(\gtrsim)} {} & f_b(\mathsf{lab}(s_1, \alpha), \ldots, \mathsf{lab}(t', \alpha), \ldots, \mathsf{lab}(s_n, \alpha)) \\
={} & \mathsf{lab}(t, \alpha)
\end{aligned}$$

  This case concludes the proof. ◀

▶ **Lemma 52.** *If $s \to_{\mathcal{S}} t$ then $\mathsf{lab}(s, \alpha) \to_{\mathcal{S}_{\mathsf{lab}}/\mathcal{D}\mathsf{ec}(\gtrsim)} \mathsf{lab}(t, \alpha)$ for all assignments α.*

Proof. The same argument as Lemma 51 goes through. ◀

Proof of Theorem 7. The "if" direction follows from Lemmata 51 and 52. In order to show the "only if" direction we consider the contraposition. Suppose that $\mathcal{R}_{\mathsf{lab}}/(\mathcal{S}_{\mathsf{lab}} \cup \mathcal{D}\mathsf{ec}(\gtrsim))$ has an infinite rewrite sequence. Unlabeling each term in the sequence, we obtain an infinite rewrite sequence of $\mathcal{R}/\mathcal{S}$. ◀

B Omitted Proofs in Section 6

Lemma 33 claims that if $s \succ_{\mathcal{G}} t$ then $s{\downarrow} \succ_{\mathsf{mpo}} t{\downarrow}$. This follows from that $s \trianglerighteq t$ and $\mathsf{root}(t) \in \mathcal{F} \setminus \mathcal{G}$ imply $s{\downarrow} \trianglerighteq t{\downarrow}$.

▶ **Lemma 53.** *If $\mathsf{root}(t) \in \mathcal{F} \setminus \mathcal{G}$, $s \to_{\mathsf{F}(\mathcal{G})} s'$, and $s \trianglerighteq t$ then $\mathsf{root}(t') \in \mathcal{F} \setminus \mathcal{G}$, $t \to_{\mathsf{F}(\mathcal{G})}^{=} t'$, and $s' \trianglerighteq t'$ for some t'.*

Proof. Suppose $s \xrightarrow{p}_{\mathsf{F}(\mathcal{G})} s'$ and $s \trianglerighteq t$. We perform induction on p.

- If $s = t$ then $p > \epsilon$ because of $\mathsf{root}(t) \in \mathcal{F} \setminus \mathcal{G}$. By taking $t' = s'$ the claim holds.
- If $p = \epsilon$ then $s = \ell\sigma$ and $s' = r\sigma$ for some rule $\ell \to r \in \mathsf{F}(\mathcal{G})$. By assumption we have $\mathsf{root}(t) \notin \mathcal{F}\mathsf{un}(\ell)$, so there exists $x \in \mathcal{V}\mathsf{ar}(\ell)$ such that $x\sigma \trianglerighteq t$. Since $\mathcal{V}\mathsf{ar}(\ell) = \mathcal{V}\mathsf{ar}(r)$ holds, we obtain $s' \trianglerighteq x\sigma$.
- Otherwise, s is of form $f(s_1, \ldots, s_n)$, $s_i \trianglerighteq t$, and $p = jq$ for some indexes $1 \leqslant i, j \leqslant n$ and position q.
  - If $i = j$ then by the induction hypothesis we obtain $t \to_{\mathsf{F}(\mathcal{G})}^{=} t'$ and $s'|_i \trianglerighteq t'$ for some t'.
  - If $i \neq j$ then by setting $t' = t$ we have $t \to_{\mathsf{F}(\mathcal{G})}^{=} t'$ and $s'|_i = s_i \trianglerighteq t'$.

  In either case t' is a subterm of s'. Therefore, the claim holds. ◀

▶ **Lemma 54.** *If $s \trianglerighteq t$ and $\mathsf{root}(t) \in \mathcal{F} \setminus \mathcal{G}$ then $s{\downarrow} \trianglerighteq t{\downarrow}$.*

Proof. Suppose $s \trianglerighteq t$ and $s \to^n_{\mathsf{F}(\mathcal{G})} s{\downarrow}$. By using Lemma 54 n times we derive $s{\downarrow} \trianglerighteq t'$ for some term t' with $t \to^*_{\mathsf{F}(\mathcal{G})} t'$. Since subterms of flattened terms are flattened, t' is flattened. Hence, $s{\downarrow} \trianglerighteq t' = t'{\downarrow} = t{\downarrow}$. ◀

Proof of Lemma 33. Suppose $s \succ_{\mathcal{G}} t$ with $s = f(s_1, \ldots, s_m)$. We have $f \in \mathcal{F} \setminus \mathcal{G}$ and $s{\downarrow} = f(s_1{\downarrow}, \ldots, s_n{\downarrow})$. By induction on the derivation of $s \succ_{\mathcal{G}} t$ we show $s{\downarrow} \succ_{\mathsf{mpo}} t{\downarrow}$.

(1) If $s \succ_{\mathcal{G}} t$ is derived by Definition 17(2) then $t = g(t_1, \ldots, t_n)$, $f \succ g$, and $s \succ_{\mathcal{G}} t_j$ for all j. Since $f \succ g$ guarantees $f \in \mathcal{F} \setminus \mathcal{G}$, the identity $s{\downarrow} = f(s_1{\downarrow}, \ldots, s_m{\downarrow})$ holds. Moreover, the induction hypothesis yields $s{\downarrow} \succ_{\mathcal{G}} t_j{\downarrow}$ for all j. Thus, the inequalities

$$s{\downarrow} = f(s_1{\downarrow}, \ldots, s_m{\downarrow}) \succ_{\mathsf{mpo}} g(t_1{\downarrow}, \ldots, t_n{\downarrow}) \succsim_{\mathsf{mpo}} t{\downarrow}$$

follow by MPO (2) and Lemma 34.

(2) If $s \succ_{\mathcal{G}} t$ is derived by Definition 17(1) then $s_i \trianglerighteq t$. We distinguish three subcases on the shape of t.

- If t is a variable then $t \in \mathcal{V}\mathsf{ar}(s_i) = \mathcal{V}\mathsf{ar}(s_i{\downarrow})$. As we have $s{\downarrow} \vartriangleright t{\downarrow}$. Thus, the claim follows by Definition 17(1).
- If $t = g(t_1, \ldots, t_n)$ and $g \in \mathcal{F} \setminus \mathcal{G}$ then Lemma 54 yields $s_i{\downarrow} \trianglerighteq t{\downarrow}$, which leads to $s{\downarrow} \vartriangleright t{\downarrow}$. Thus, the claim follows by Definition 17(1).
- If $t = g(t_1, \ldots, t_n)$ and $g \in \mathcal{G} \cup \mathcal{F}_{\mathsf{c}}$ then $s \vartriangleright t_j$ for all j. So $s \succ_{\mathcal{G}} t_j$ holds for all j. Therefore, the proof for case (1) goes through. ◀

Lemma 39 claims that if s is ground and $s \trianglerighteq t$ then $s{\downarrow} \succsim_{\mathsf{mpo}} t{\downarrow}$. To facilitate its inductive proof, we show the following lemma.

▶ **Lemma 55.** *Let $C = f(s_1, \ldots, s_{i-1}, \Box, s_{i+1}, \ldots, s_n)$. If $C[t]$ is ground then $C[t]{\downarrow} \succsim_{\mathsf{mpo}} t{\downarrow}$.*

Proof. Suppose $C[t]$ is ground. We perform induction on $C[t]$ with respect to $\to^+_{\mathsf{F}(\mathcal{G})}$. If $t{\downarrow} = \mathsf{c}_0$ then $C[t]{\downarrow} \succsim_{\mathsf{mpo}} t{\downarrow}$ holds because $C[t]{\downarrow}$ is ground and c_0 is the minimum ground term. Suppose $t{\downarrow} \neq \mathsf{c}_0$. We proceed with analyzing f and whether s_j are flattened.

1. If $f \in \mathcal{F} \setminus \mathcal{G}$ then $C[t]{\downarrow} = f(s_1{\downarrow}, \ldots, t{\downarrow}, \ldots, s_n{\downarrow})$. So $C[t]{\downarrow} \succsim_{\mathsf{mpo}} t{\downarrow}$ is obtained by MPO (1).
2. If $f \in \mathcal{G}$ then take $D = \mathsf{c}_n(s_1, \ldots, s_{i-1}, \Box, s_{i+1}, \ldots, s_n)$. We have $C[t] \to_{\mathsf{F}(\mathcal{G})} D[t]$. Thus, we obtain $C[t]{\downarrow} = D[t]{\downarrow} \succsim_{\mathsf{mpo}} t{\downarrow}$ by the induction hypothesis.
3. Similarly, if $s_j \to^+_{\mathsf{F}(\mathcal{G})} s_j{\downarrow}$ for some $j \neq i$, take $D = \mathsf{c}_n(s_1{\downarrow}, \ldots, s_{i-1}{\downarrow}, \Box, s_{i+1}{\downarrow}, \ldots, s_n{\downarrow})$. The same argument applies.
4. If $f \in \mathcal{F}_{\mathsf{c}}$ and $s_j = \mathsf{c}_m(u_1, \ldots, u_n)$ for some $j \neq i$, by taking the context

$$D = \begin{cases} \mathsf{c}_n(s_1, \ldots, s_{j-1}, u_1, \ldots, u_m, s_{j+1}, \ldots, s_{i-1}, C', s_{i+1}, \ldots, s_n) & \text{if } j < i \\ \mathsf{c}_n(s_1, \ldots, s_{i-1}, C', \ldots, s_{i+1}, s_{j-1}, u_1, \ldots, u_m, s_{j+1}, \ldots, s_n) & \text{if } j > i \end{cases}$$

   the claim is verified as in the last two cases.
5. Otherwise, $f = \mathsf{c}_n$ and $s_j{\downarrow} = s_j$ for all $1 \leqslant i \leqslant m$. If $n = 1$, we immediately obtain $C[t]{\downarrow} = t{\downarrow}$. So hereafter we assume $n \geqslant 2$. Furthermore we distinguish two cases, depending on $t{\downarrow}$.
   - If $t{\downarrow} = \mathsf{c}_m(t_1, \ldots, t_n)$ then $m \neq 1$. As $t{\downarrow} \neq \mathsf{c}_0$, we have $m \geqslant 2$. Thus, $m + n - 1 > m$ holds. Therefore, by MPO (3) the inequality

     $$C[t]{\downarrow} = \mathsf{c}_{m+n-1}(s_1, \ldots, s_{i-1}, u_1, \ldots, u_m, s_{i+1}, \ldots, s_n) \succ_{\mathsf{mpo}} t{\downarrow}$$

     is derived.
   - Otherwise, $C[t]{\downarrow} = \mathsf{c}_{m+n-1}(s_1, \ldots, t{\downarrow}, \ldots, s_n) \succ_{\mathsf{mpo}} t{\downarrow}$ follows by MPO (1). ◀

Proof of Lemma 39. The claim is shown by induction on s together with Lemma 55. ◀