ELSEVIER Theoretical Computer Science 200 (1998) 289–312

Theoretical Computer Science

# Operational semantics of rewriting with priorities

Jaco van de Pol*

*Eindhoven University of Technology, Department of Computing Science, P.O. box 513, 5600 MB Eindhoven, Netherlands*

Received March 1996; revised May 1997
Communicated by U. Montanari

**Abstract**

We study the semantics of term rewriting systems with rule priorities (PRS), as introduced in Baeten et al. (1989). Three open problems posed in that paper are solved, by giving counter examples. Moreover, a class of executable PRSs is identified. A translation of PRSs into transition system specifications (TSS) is given. This translation introduces negative premises. We prove that the translation preserves the operational semantics. © 1998—Elsevier Science B.V. All rights reserved

*Keywords:* Term rewriting systems; Rule priorities; Structural operational semantics; Transition system specification; Negative premises

## 1. Introduction

### 1.1. Motivation

In [1], term rewriting with rule priorities has been introduced. A *priority rewrite system* (PRS) extends an ordinary term rewriting system (TRS) with a partial order on the rules. The main idea is to resolve a conflict between two rules by giving priority to the largest rule. One may hope that by ordering the rules of a non-confluent TRS, a confluent PRS can be obtained (i.e. a system in which each reduction eventually gives the same result). Indeed, some results of this kind are known.

The above motivation of the priority mechanism can be seen as an implementation issue: priorities drastically decrease the amount of non-determinism involved in term rewriting.

The second motivation evolves from a specification point of view. The priority mechanism adds expressive power. We mention two points only: In a signature containing

* E-mail: jaco@win.tue.nl.

*PII* S0304-3975(97)00283-1

the booleans, an equality predicate for an arbitrary sort can be specified by two rules only (see Example 34). This cannot be done in ordinary TRSs. The other indication is: The one step reduction relation of PRSs is not decidable in general.

These motivations justify the mathematical study of the priority mechanism itself. In this paper, we will not be concerned with restrictions on the rules, the partial order or the reduction strategy. Such restrictions can be fruitful, but form a different topic. We mention the following restrictions: specificity order on rules; left linear rules; leftmost/innermost reduction or a lazy strategy; operator-constructor discipline. See e.g. [4–6] for various results obtained by making such restrictions.

### 1.2. Contribution

The semantics of a PRS is not straightforward. The reason is that the question whether a certain rule may be applied, cannot be answered by syntactically matching the rules of higher priority (cf. Example 3). It is even the case that not every PRS will have a semantics.

In [1], a PRS is called meaningful if it has a so-called *unique sound and complete rewrite set*. A certain monotonic operator is associated to a PRS, which reaches its least and greatest fixed points at some closure ordinal $\alpha$. It has been proved that in case these fixed points coincide, the PRS is meaningful. It has also been shown that if the PRS is bounded, then the least and greatest fixed points are equal. Three open questions concerning this fixed point construction were posed:

(I) Is the associated monotonic operator always continuous?

(II) Is the closure ordinal $\alpha$ always finite?

(III) Is coincidence of the least and the greatest fixed point a necessary condition for being meaningful?

We solve these questions in a negative way, i.e. by giving a counterexample to each of them (Section 2.4). We also give a sufficient condition for decidability of the one-step reduction relation. This can be used to identify a subclass of executable PRSs (Section 2.3), addressing another question posed in [1]. In particular, the one step reduction relation of the PRS is decidable, if the underlying TRS is strongly normalizing.

In Section 3.2, we give a translation of a PRS into a transition system specification (TSS) with negative premises [2, 7]. Such a specification can be seen as an inductive definition with negative premises. Such definitions are not always meaningful. We show (Theorems 39 and 42) that the operational semantics is preserved under this translation. Another application of TSS theory to term rewriting occurs in [3].

This translation relates the semantics of priority rewriting given in [1] with general techniques to deal with negation in operational semantics and logic programming (for references to logic programming we refer to [7]). It also explains the negative answer to the third of the open questions. Finally, it opens the way to combining priorities with positive/negative conditions.

## 2. Term rewriting with rule priorities

In Sections 2.1 and 2.2, we shortly recapitulate the definitions and some theory on priority rewrite systems (PRSs). These sections are based on [1]; only Example 5 is new. In Section 2.3 we identify a subclass of executable PRSs and Section 2.4 contains counter examples to some open questions posed in [1].

### 2.1. *Definition and semantics*

We assume a signature $\Sigma$ of the form $(\mathscr{F},\mathscr{V})$. Here $\mathscr{F}$ is a set of function symbols with fixed arities, $\mathscr{V}$ is an infinite set of variables. Sets of (open) terms $\mathscr{T}(\mathscr{F},\mathscr{V})$ and closed terms $\mathscr{T}(\mathscr{F})$ are defined as usual. Var($s$) denotes the variables occurring in term $s$. A substitution is a finite function from variables to terms.

**Definition 1** (*Baeten et al.* [1, Definition 2.5]). (1) A *rule* is a pair of terms, written $l \mapsto r$, such that $l$ is not a variable and the variables of $r$ occur in $l$.

(2) A *term rewriting system* (TRS) is a pair $(\Sigma,R)$, with $R$ a set of rules.

(3) A *priority rewrite system* (PRS) is a tuple $(\mathscr{R}, >)$, with $\mathscr{R}$ a TRS, and $>$ a partial order on the rules of $\mathscr{R}$.

In examples, the priority ordering will be denoted by arrows. We call $\mathscr{R}$ the *underlying TRS* of a PRS $(\mathscr{R}, >)$.

**Definition 2.** Let PRS $\mathscr{P} = (\mathscr{R}, >)$ be given.

(1) Let $r$ be a rule in $\mathscr{P}$. An *r-rewrite* (written $s \mapsto^r t$) is a closed instance of $r$.

(2) Let $R$ be a set of rewrites. The closure of $R$ under closed context is denoted by $\to_R$. The reflexive transitive closure of $\to_R$ is denoted by $\twoheadrightarrow_R$. With $s \twoheadrightarrow_R^{\text{int}} t$ we denote an *internal* reduction, i.e. a reduction where each contracted redex is not at the root. If $R$ is a large expression, we also write $R \models s \to t$, $R \models s \twoheadrightarrow t$ etc.

(3) We write $\to_{\mathscr{R}}$, $\twoheadrightarrow_{\mathscr{R}}$ and $\twoheadrightarrow_{\mathscr{R}}^{\text{int}}$ in case we work in the underlying TRS. (i.e. the reductions may use all rewrites).

The priorities are used to indicate preference of one rule above another. In this way a conflict between two rules can be resolved. So not every rewrite is *enabled*. A rewrite is only enabled, if it is not blocked by a rule of higher priority. Let us look at an example before making this formal.

**Example 3** (*Baeten et al.* [1, Example 2.1]).

$$\boxed{\begin{array}{lll} & r1: & P(0) \mapsto 0 \\ & r2: & P(S(x)) \mapsto x \\ \Big\downarrow & r3: & x+0 \mapsto x \\ & r4: & x+y \mapsto S(x+P(y)) \end{array}}$$

The rewrite $x+0 \mapsto^{r4} S(x+P(0))$ is blocked, because $r3$ takes precedence. However, also $x+(P(S(0))) \mapsto^{r4} S(x+P(P(S(0))))$ should be blocked by $r3$, because *eventually*, $P(S(0))$ becomes 0. The correct reduction is: $x+(P(S(0))) \mapsto^{r2} x+0 \mapsto^{r3} x$.

As Example 3 shows, the definition of the reduction relation induced by a PRS is not straightforward. The rewrite $s+t \mapsto^{r4} S(s+P(t))$ is enabled only, if $t \not\twoheadrightarrow 0$. So in the definition of the one step reduction relation, the *negation* of the more step reduction relation occurs. This explains the following definition.

**Definition 4** (*Baeten et al.* [1, Definitions 2.8 and 2.9]). Let PRS $\mathscr{P}=(\mathscr{R},>)$ be given, with a rewrite set $R$.

(1) Let $x=s \mapsto^{r} t$ be a rewrite of $\mathscr{P}$. $R$ is an *obstruction* for $x$ (written $x \lhd R$) if there is a rewrite $s' \mapsto^{r'} t'$ of $\mathscr{P}$ with $r'>r$ and a reduction $s \twoheadrightarrow_R^{\text{int}} s'$, using *precisely all* rewrites in $R$.

(2) A rewrite $x$ of $\mathscr{P}$ is *correct* with respect to $R$, if there is no obstruction $O \subseteq R$ for $x$.

(3) $R$ is *sound* if all its rewrites are correct w.r.t. $R$.

(4) $R$ is *complete* if it contains all rewrites of $\mathscr{P}$ that are correct w.r.t. $R$.

(5) $\mathscr{P}$ is *meaningful* if it has a unique sound and complete rewrite set. This set is the *semantics* of $\mathscr{P}$.

In [1] an example of a PRS is given that does not have a sound and complete rewrite set (see Example 44), as well as a PRS that has more than one sound and complete rewrite set. Neither of them is meaningful by Definition 4.5. The following example will also play a rôle in Section 2.4.

**Example 5.** Consider the following PRS $\mathscr{P}$ with a constant $a$ and a unary function symbol $f$:

$$\boxed{\left\downarrow \begin{array}{l} f(a) \mapsto f(f(f(a))) \\ f(x) \mapsto a \end{array}\right.}$$

We write $f^n(a)$ for the $n$-fold application of $f$ to $a$. Note that all closed terms are of the form $f^n(a)$. We claim that $\mathscr{P}$ is meaningful, because the following set is the unique sound and complete rewrite set for it:

$$R := \{f(a) \mapsto f^3(a)\} \cup \{f^{2m+2}(a) \mapsto a \mid m \geqslant 0\}.$$

*Completeness*: The only rewrites not in $R$ are of the form $f^{2m+1}(a) \mapsto a$, for some $m \geqslant 0$, but these are not correct with respect to $R$, because $f^{2m+1}(a) \twoheadrightarrow_R^{\text{int}} f(a)$. (If

$m=0$ in 0 steps, if $m>0$ in one step). So $R$ contains all rewrites that are correct w.r.t. itself.

*Soundness*: Note that if $s$ has an even number of $f$-symbols, and $s \twoheadrightarrow_R t$, then $t$ has an even number of $f$-symbols too. So for no $m$ we have $f^{2m+2}(a) \twoheadrightarrow_R^{\text{int}} f(a)$, hence all rewrites of $R$ are correct w.r.t. itself.

*Uniqueness*: Let $S$ be a sound and complete rewrite set. By completeness $S$ contains $f(a) \mapsto f^3(a)$, so for all $m \geqslant 0$, $f(a) \twoheadrightarrow_S f^{2m+1}(a)$. Assume, towards a contradiction, that $S$ contains $f^{2m+1}(a) \mapsto a$ for some $m \geqslant 0$. Then $f(a) \twoheadrightarrow_S a$; hence also $f^{2m}(a) \twoheadrightarrow_S a$. Now by soundness, $f^{2m+1}(a) \mapsto a$ is not in $S$: contradiction. This shows $S \subseteq R$. Vice versa, let $x \in R$, then $x$ is correct w.r.t. $R$ (soundness of $R$), hence also correct w.r.t. the subset $S$ and, by completeness of $S$, $x \in S$. Hence $S=R$, proving uniqueness.

### 2.2. Fixed points

In Example 5 a rewrite set was given in advance and then checked for soundness and completeness. We want of course a method to compute this set by means of successive approximations. This is the aim of this section.

**Definition 6** (*Baeten et al.* [1, Definitions 2.13 and 3.2]). (1) Let $R$ be a set of rewrites of PRS $\mathscr{P}$. Then the *closure* of $R$, written $R^{\mathscr{P}}$ consists of all rewrites that are correct w.r.t. $R$.

(2) Put $\mathbf{T}_{\mathscr{P}}(R) := (R^{\mathscr{P}})^{\mathscr{P}}$.

**Lemma 7** (Baeten et al. [1, Lemma 2.14]). *Let $R$ be a set of rewrites for PRS $\mathscr{P}$.*
(1) *$R$ is sound* $\Leftrightarrow R \subseteq R^{\mathscr{P}}$.
(2) *$R$ is complete* $\Leftrightarrow R \supseteq R^{\mathscr{P}}$.
(3) $R \subseteq S \Rightarrow R^{\mathscr{P}} \supseteq S^{\mathscr{P}}$.

Combining (1) and (2) of this lemma, we see that we need a unique fixed point of the closure map $(\,)^{\mathscr{P}}$. Unfortunately, this map is not monotonic, but antitonic, as seen from the last part of the lemma. But then the operation $\mathbf{T}_{\mathscr{P}}$ is monotonic, so we can compute its least and greatest fixed points. Consider the following construction, parameterized by an arbitrary PRS $\mathscr{P}$. (Here and in the sequel, $\alpha$ ranges over arbitrary ordinals and $\lambda$ over limit ordinals; $m$ and $n$ range over finite ordinals.)

**Definition 8** (*Baeten et al.* [1, Definition 3.3]).

$$\begin{aligned}
&\mathbf{T}_{\mathscr{P}}\uparrow 0 := \emptyset, && \mathbf{T}_{\mathscr{P}}\downarrow 0 := \emptyset^{\mathscr{P}},\\
&\mathbf{T}_{\mathscr{P}}\uparrow(\alpha+1) := \mathbf{T}_{\mathscr{P}}(\mathbf{T}_{\mathscr{P}}\uparrow\alpha), && \mathbf{T}_{\mathscr{P}}\downarrow(\alpha+1) := \mathbf{T}_{\mathscr{P}}(\mathbf{T}_{\mathscr{P}}\downarrow\alpha),\\
&\mathbf{T}_{\mathscr{P}}\uparrow\lambda := \bigcup_{\alpha<\lambda}(\mathbf{T}_{\mathscr{P}}\uparrow\alpha), && \mathbf{T}_{\mathscr{P}}\downarrow\lambda := \bigcap_{\alpha<\lambda}(\mathbf{T}_{\mathscr{P}}\downarrow\alpha).
\end{aligned}$$

**Proposition 9** (Baeten et al. [1, Theorem 3.5]). *For all PRSs $\mathscr{P}$ and ordinals $\alpha$,*
(1) $(\mathbf{T}_{\mathscr{P}}\uparrow\alpha)^{\mathscr{P}} = \mathbf{T}_{\mathscr{P}}\downarrow\alpha$.
(2) $(\mathbf{T}_{\mathscr{P}}\downarrow\alpha)^{\mathscr{P}} = \mathbf{T}_{\mathscr{P}}\uparrow(\alpha+1)$.

**Proposition 10** (Baeten et al. [1, Proposition 3.8]). *For all PRSs $\mathscr{P}$ and ordinals $\alpha$,*
(1) $\mathbf{T}_{\mathscr{P}}\uparrow\alpha$ *is sound.*
(2) $\mathbf{T}_{\mathscr{P}}\downarrow\alpha$ *is complete.*
(3) *If $R$ is sound and complete, then* $\mathbf{T}_{\mathscr{P}}\uparrow\alpha \subseteq R \subseteq \mathbf{T}_{\mathscr{P}}\downarrow\alpha$.

**Proof.** (1) and (2) are proved in [1]. Part 3 is not explicitly mentioned there, although it is needed in the following corollary.

Assume that $R$ is sound and complete, then $R = R^{\mathscr{P}}$ by Proposition 7 (1) and (2), hence $\mathbf{T}_{\mathscr{P}}(R) = R$. With induction to $\alpha$, we prove that $\mathbf{T}_{\mathscr{P}}\uparrow\alpha \subseteq R$:
- $\mathbf{T}_{\mathscr{P}}\uparrow 0 = \emptyset \subseteq R$;
- If $\mathbf{T}_{\mathscr{P}}\uparrow\alpha \subseteq R$, then as $\mathbf{T}_{\mathscr{P}}$ is monotonic, we have $\mathbf{T}_{\mathscr{P}}\uparrow(\alpha+1) = \mathbf{T}_{\mathscr{P}}(\mathbf{T}_{\mathscr{P}}\uparrow\alpha) \subseteq \mathbf{T}_{\mathscr{P}}(R) = R$;
- If $\mathbf{T}_{\mathscr{P}}\uparrow\alpha \subseteq R$ for all $\alpha < \lambda$, then also $\mathbf{T}_{\mathscr{P}}\uparrow\lambda = \bigcup_{\alpha<\lambda}\mathbf{T}_{\mathscr{P}}\uparrow\alpha \subseteq R$.

Then by Proposition 7.3, $\mathbf{T}_{\mathscr{P}}\downarrow\alpha = (\mathbf{T}_{\mathscr{P}}\uparrow\alpha)^{\mathscr{P}} \supseteq R^{\mathscr{P}} = R$. □

Because the operation $\mathbf{T}_{\mathscr{P}}$ is monotonic, it has a least fixed point, which is reached at some ordinal. We define the *closure ordinal* of a PRS $\mathscr{P}$ as the first $\alpha$ such that $\mathbf{T}_{\mathscr{P}}\uparrow\alpha = \mathbf{T}_{\mathscr{P}}\uparrow(\alpha+1)$. Note that for the closure ordinal also $\mathbf{T}_{\mathscr{P}}\downarrow\alpha = \mathbf{T}_{\mathscr{P}}\downarrow(\alpha+1)$. In this way we find the least and the greatest fixed points for the map $\mathbf{T}_{\mathscr{P}}$. We now have the following corollary:

**Corollary 11** (Baeten et al. [1, Corollary 3.9]). *Let $\alpha$ be the closure ordinal of a PRS $\mathscr{P}$. If* $\mathbf{T}_{\mathscr{P}}\uparrow\alpha = \mathbf{T}_{\mathscr{P}}\downarrow\alpha$, *then $\mathscr{P}$ is meaningful and* $\mathbf{T}_{\mathscr{P}}\uparrow\alpha$ *is its semantics.*

Example 24 – which can be read independently of the next section – shows, that the condition of the corollary is not a necessary one.

### 2.3. *An executable class of PRSs*

In this section, we will prove that locally finite PRSs have a closure ordinal at most $\omega$. Consequently, given a bounded PRS with finitely many rules, we can actually compute the finite set of $\rightarrow$-successors of each term $s$. In this sense, the PRS can be executed as a program with input $s$. This answers a question put in [1], by giving a reasonable class of PRSs that is *executable.*

We first define some relevant properties of PRSs, in terms of the underlying TRS.

**Definition 12.** (1) A TRS is *strongly normalizing* if all reduction sequences are finite.
(2) A possibly infinite reduction sequence $s_0 \rightarrow s_1 \rightarrow \cdots$ is *bounded* if there exists an $n$ such that for all $i$, $|s_i| \leqslant n$. (Here $|s|$ denotes the length of a term $s$ in symbols).

(3) A TRS $\mathscr{R}$ is *bounded* if all reductions sequences in $\mathscr{R}$ are bounded.
(4) A TRS is *locally finite* if for all $s$, the set $\{t \mid s \twoheadrightarrow t\}$ is finite.
(5) A PRS is bounded (locally finite) if its underlying TRS is.

An easy syntactic check for boundedness is that all rules are "non-duplicating" and "non-length-increasing". The first property holds if the multiset of variables on the right hand side is contained in the multiset of variables on the left. A rule is non-length-increasing if its right hand side contains not more symbols than its left hand side. (One can even assign weights to the function symbols). The existence of a recursive path order also implies boundedness, as strong normalization is stronger than boundedness. None of these syntactic conditions is necessary, however.

**Proposition 13.** *Let $\mathscr{P}$ be a bounded PRS with closure ordinal $\alpha$. Then $\mathbf{T}_{\mathscr{P}}\uparrow\alpha = \mathbf{T}_{\mathscr{P}}\downarrow\alpha$.*

**Proof.** This follows immediately from Propositions 3.11 and 3.14 in [1]. □

It will be shown that if the set of rules is finite, then $\alpha$ is at most $\omega$ (Proposition 16). We first need Proposition 14, relating the properties defined above, and the auxiliary Lemma 15.

**Proposition 14.** *Let $\mathscr{R} = (\Sigma, R)$ be a TRS.*
*(1) If $\mathscr{R}$ is strongly normalizing, then $\mathscr{R}$ is bounded.*
*(2) If $\mathscr{R}$ is locally finite, then $\mathscr{R}$ is bounded.*
*(3) If $R$ is finite and $\mathscr{R}$ is bounded, then $\mathscr{R}$ is locally finite.*

**Proof.** (1) Given a sequence $s_0 \rightarrow s_1 \rightarrow \cdots$, we can take the length of the largest term in it, as the sequence must be finite.
(2) Given a sequence $s_0 \rightarrow s_1 \rightarrow \cdots$, we can take the length of the largest term in the finite set $\{t \mid s_0 \twoheadrightarrow t\}$.
(3) Suppose $\mathscr{R}$ is bounded and finite; let $s$ be given. Put $V = \{t \mid s \twoheadrightarrow_{\mathscr{R}} t\}$ and $E := \rightarrow_{\mathscr{R}} \cap\, V \times V$. Consider $(V, E)$ as an $s$-rooted graph.
  (a) $(V, E)$ is finitely branching because $\rightarrow_{\mathscr{R}}$ is. This is because the set of rules is finite, hence every term contains only finitely many redexes.
  (b) All acyclic paths in $(V, E)$ are finite. This is because each path corresponds with a reduction sequence in $\twoheadrightarrow_{\mathscr{R}}$. By boundedness, all terms in this sequence are shorter than $n$ for some $n$. Furthermore, these terms are built from a finite set of function symbols: those occurring in $s$ or in the finite set of rules. So there are only finitely many different terms on each path.
Now (a) and (b) imply that $V$ is finite. To see this, we apply König's Lemma on an acyclic subgraph of $(V, E)$ that covers all nodes in $V$. To obtain such a graph, we proceed as follows.

For $t \in V$, let $d(t)$ denote the distance from the root $s$ to $t$. Define $D \subseteq E$ as $\{(r,t) \mid d(r)+1=d(t)\}$. Then $D$ is acyclic by construction. For each $t \in V$, we have $sD^*t$ as can be shown by induction on $d(t)$. □

**Lemma 15.** *Let $\mathscr{P}$ be a locally finite PRS. Let $x$ be a rewrite of $\mathscr{P}$. Put $V := \bigcup\{O \mid x \lhd O\}$. Then $V$ is finite.*

**Proof.** Because $\mathscr{P}$ is locally finite, the set $\{t \mid s \twoheadrightarrow t\}$ is finite for each $s$. Each term has finitely many subterms, so the set $\{t \mid \exists C, r. s \twoheadrightarrow r \wedge C[t]=r\}$ is also finite for each $s$. Now each $a \mapsto b$ in $V$ is in an obstruction, so for some context $C$, we have $\mathrm{lhs}(x) \twoheadrightarrow^{\mathrm{int}} C[a] \rightarrow C[b]$. Hence $V$ is finite. □

**Proposition 16.** *If $\mathscr{P}$ is a locally finite PRS then its closure ordinal is at most $\omega$.*

**Proof.** It is enough to prove that $\mathbf{T}_{\mathscr{P}}\uparrow\omega \supseteq \mathbf{T}_{\mathscr{P}}\uparrow(\omega+1)$. Consider $x \in \mathbf{T}_{\mathscr{P}}\uparrow(\omega+1) = (\mathbf{T}_{\mathscr{P}}\downarrow\omega)^{\mathscr{P}}$. Put $V := \bigcup\{O \mid x \lhd O\}$. Because $x$ is correct w.r.t. $\mathbf{T}_{\mathscr{P}}\downarrow\omega$, there is no obstruction of $x$ entirely in $\mathbf{T}_{\mathscr{P}}\downarrow\omega$, so we can find a set $W \subseteq V$, such that $W \cap \mathbf{T}_{\mathscr{P}}\downarrow\omega = \emptyset$ and for each obstruction $O$ of $x$, $W \cap O \neq \emptyset$. By Lemma 15, $V$ is finite, so $W$ is finite too. Therefore, there exists a $n$, such that $W \cap \mathbf{T}_{\mathscr{P}}\downarrow n = \emptyset$. But then $x \in \mathbf{T}_{\mathscr{P}}\uparrow(n+1)$, so $x \in \mathbf{T}_{\mathscr{P}}\uparrow\omega$. □

**Corollary 17.** *If $\mathscr{P}$ is a locally finite PRS, then $\mathbf{T}_{\mathscr{P}}\uparrow\omega$ is its semantics.*

**Proof.** By Lemma 13 $\mathscr{P}$ has a semantics, which must have been reached at $\omega$ by Proposition 16. □

As a corollary we have that bounded PRSs with finitely many rules are executable in the sense that for each term $s$, the set of $\rightarrow$-successors is finite and computable.

**Theorem 18.** *Let $\mathscr{P}$ be a bounded PRS with finitely many rules. Then $\mathscr{P}$ is executable.*

**Proof.** By Proposition 14.3, $\mathscr{P}$ is locally finite, hence (by the previous corollary) the semantics of $\mathscr{P}$ is $\mathbf{T}_{\mathscr{P}}\uparrow\omega$. So given a closed term $s$, we have to compute the finite set $\{t \mid \mathbf{T}_{\mathscr{P}}\uparrow\omega \models s \rightarrow t\}$.

This is done by generating all successors $t$ of $s$ in the underlying TRS, and then testing whether the rewrite $x$ used to obtain $t$ is enabled, i.e. whether $x \in \mathbf{T}_{\mathscr{P}}\uparrow\omega$. Note that if so, then it is contained in $\mathbf{T}_{\mathscr{P}}\uparrow n$ for some finite $n$ already. Otherwise, it is not in $\mathbf{T}_{\mathscr{P}}\downarrow\omega$ either, hence it is outside $\mathbf{T}_{\mathscr{P}}\downarrow n$ for some finite $n$ already. So we consider the sequence $\mathbf{T}_{\mathscr{P}}\uparrow 0, \mathbf{T}_{\mathscr{P}}\downarrow 0, \mathbf{T}_{\mathscr{P}}\uparrow 1, \mathbf{T}_{\mathscr{P}}\downarrow 1, \ldots$ until we find an $n$, with $x \in \mathbf{T}_{\mathscr{P}}\uparrow n$ or $x \notin \mathbf{T}_{\mathscr{P}}\downarrow n$.

We still need to prove that for all finite $n$, it is decidable whether $s \mapsto^{r} t$ is in $\mathscr{P}^n(\emptyset)$ (the $n$-fold application of $(\,)^{\mathscr{P}}$). This is proved with induction to $n$. For $n=0$, the answer is clearly NO. Now suppose that for some $n$, $\mathscr{P}^n(\emptyset)$ is decidable. Let some rewrite $s \mapsto^r t$ be given. It is in $\mathscr{P}^{n+1}(\emptyset)$ if and only if it is correct w.r.t. $\mathscr{P}^n(\emptyset)$. This is the case if and only if there is no rewrite $s' \mapsto^{r'} t'$ with $r' > r$ and $\mathscr{P}^n \models s \twoheadrightarrow^{\text{int}} s'$. This can be tested by generating all terms reachable from $s$ using $\twoheadrightarrow^{\text{int}}_{\mathscr{R}}$ (there are only finitely many because $\mathscr{P}$ is locally finite), and test whether the used rewrites are in $\mathscr{P}^n$, which is decidable by induction hypothesis. □

### 2.4. Counterexamples to open questions

In [1, p. 297] three open questions concerning the mapping $\mathbf{T}_{\mathscr{P}}$ are posed

(I) Is the mapping $\mathbf{T}_{\mathscr{P}}$ always continuous, instead of only monotonic?

(II) Is the closure ordinal of each PRS finite?

(III) Is the condition of Corollary 11 necessary? That is, does every meaningful PRS $\mathscr{P}$ with closure ordinal $\alpha$, satisfy $\mathbf{T}_{\mathscr{P}}\uparrow\alpha = \mathbf{T}_{\mathscr{P}}\downarrow\alpha$?

We have found counterexamples to each of these questions. First, Example 19 provides for a finite PRS, with closure ordinal $\omega$. This is a counterexample to (II). It is easy to extend this example in order to find a closure ordinal beyond $\omega$ (Example 22). This refutes (I), because if $\mathbf{T}_{\mathscr{P}}$ were continuous, the closure ordinal would be at most $\omega$. Finally, we show that for the PRS $\mathscr{P}$ of Example 5, $\mathbf{T}_{\mathscr{P}}\uparrow\alpha \neq \mathbf{T}_{\mathscr{P}}\downarrow\alpha$ for any $\alpha$ (see Example 24), although it is meaningful, as we already showed. This answers (III) negatively.

**Example 19.** Let $\mathscr{P}$ be the following PRS:

$$\boxed{\begin{array}{cc} & f(b) \mapsto c \\ \downarrow & f(x) \mapsto g(x) \\ & g(a) \mapsto b \\ & g(f(x)) \mapsto g(x) \end{array}}$$

Note that the TRS underlying $\mathscr{P}$ is strongly normalizing, so it is bounded. By Lemma 16, the closure ordinal is at most $\omega$. From Claim 21 below, it follows that the closure ordinal is not finite, so it must be $\omega$. This gives a negative answer to open question (II) at the beginning of this section.

**Claim 20.** *Let $R$ be an arbitrary set of rewrites. $R^{\mathscr{P}}$ contains $f^{n+1}(a) \mapsto g(f^n(a))$ if and only if $R^{\mathscr{P}} \models f^{n+1}(a) \twoheadrightarrow b$.*

**Proof.** $\Rightarrow$ is clear, because we have:

$$f^{n+1}(a) \mapsto g(f^n(a)) \mapsto g(f^{n-1}(a)) \twoheadrightarrow g(a) \mapsto b$$

$\Leftarrow$: Suppose there were a reduction from $f^{n+1}(a)$ in $R^{\mathcal{P}}$ that does not start with $g(f^n(a))$. The first step must be an innermost application of the second rule. We have to reduce the topmost $f$ at some later point. So the reduction has the following form (for some $m$, $k=n-m-1$ and $z$):

$$f^{n+1}(a) \to f^{m+1}(g(f^k(a))) \twoheadrightarrow^{\mathrm{int}} f(z) \mapsto g(z) \twoheadrightarrow b.$$

Inspection of the rules of $\mathcal{P}$ reveals that the total number of $b$-, $c$- and $g$-symbols cannot decrease during rewriting. But then the reduction above cannot exist, because $g(z)$ contains at least 2 such symbols, so it can never reduce to $b$. □

**Claim 21.** *For all $m$, the rewrite $f^{2m+1}(a) \mapsto g(f^{2m}(a))$ is in $\mathbf{T}_{\mathcal{P}}\uparrow(m+1)$, but not yet in $\mathbf{T}_{\mathcal{P}}\uparrow m$.*

**Proof.** Induction to $m$. *Base case*: because $a$ is a normal form, $\mathbf{T}_{\mathcal{P}}\downarrow 0 \models a \not\twoheadrightarrow b$, so $\mathbf{T}_{\mathcal{P}}\uparrow 1$ contains $f(a) \mapsto g(a)$; $\mathbf{T}_{\mathcal{P}}\uparrow 0 = \emptyset$.

*Induction step*: assume the claim holds for $m$, then (by Claim 20)

$$\mathbf{T}_{\mathcal{P}}\uparrow(m+1) \models f^{2m+1}(a) \twoheadrightarrow b \quad \text{and} \quad \mathbf{T}_{\mathcal{P}}\uparrow m \models f^{2m+1}(a) \not\twoheadrightarrow b,$$

hence $f^{2m+2}(a) \mapsto g(f^{2m+1}(a))$ is not contained in $\mathbf{T}_{\mathcal{P}}\downarrow(m+1)$, but it is in $\mathbf{T}_{\mathcal{P}}\downarrow m$. Therefore (Claim 20),

$$\mathbf{T}_{\mathcal{P}}\downarrow(m+1) \models f^{2m+2}(a) \not\twoheadrightarrow b \quad \text{and} \quad \mathbf{T}_{\mathcal{P}}\downarrow m \models f^{2m+2}(a) \twoheadrightarrow b.$$

Therefore, $f^{2m+3}(a) \mapsto g(f^{2m+2}(a))$ is contained in $\mathbf{T}_{\mathcal{P}}\uparrow(m+2)$, but this rewrite is not in $\mathbf{T}_{\mathcal{P}}\uparrow(m+1)$, so the claim holds for $m+1$. □

The idea of this example is that $f^m(a)$ can be reduced to $b$ for odd $m$ only. These reductions block the reductions for even $m$. The system is constructed in such a way that the larger $m$ becomes, the later we decide whether $f^m(a)$ reduces to $b$. Because the system is bounded, we cannot go beyond $\omega$. The only way to go beyond $\omega$ uses a non-bounded system. As the proof of Proposition 16 reveals, we need a term with infinitely many possible reducts. Only at stage $\omega$, the system may know that none of these is actually reached. This is the idea of the following example:

**Example 22.** Extend $\mathcal{P}$ of Example 19 with the following rules (note that the two rules are not ordered):

$$\boxed{\begin{array}{l} h(x) \mapsto f(x) \\ h(x) \mapsto h(h(h(x))) \end{array}}$$

We will show that $\mathbf{T}_{\mathcal{P}}\!\uparrow\omega \neq \mathbf{T}_{\mathcal{P}}\!\uparrow(\omega+1)$, by showing that the latter contains the rewrite $f(h(f(a))) \mapsto g(h(f(a)))$, but the former does not. Note that Claims 20 and 21 still hold for the extended system, because the proofs remain valid for the new $\mathcal{P}$.

$$\begin{array}{lll} & \text{for all } m,\ h(x) \twoheadrightarrow f^{2m+1}(x) & \text{(induction to } m\text{)} \\ \Rightarrow & \text{for all } m,\ h(f(a)) \twoheadrightarrow f^{2m+2}(a) & \\ \Rightarrow & \text{for all } m,\ \mathbf{T}_{\mathcal{P}}\!\downarrow m \models h(f(a)) \twoheadrightarrow f^{2m+2}(a) \twoheadrightarrow b & \text{(see proof of Claim 21)} \\ \Rightarrow & \text{for all } m,\ \mathbf{T}_{\mathcal{P}}\!\uparrow(m+1) \not\models f(h(f(a))) \mapsto g(h(f(a))) & \\ \Rightarrow & \mathbf{T}_{\mathcal{P}}\!\uparrow\omega \not\models f(h(f(a))) \mapsto g(h(f(a))). & \text{(defined as union)} \end{array}$$

On the other hand, $\mathbf{T}_{\mathcal{P}}\!\uparrow(\omega+1) \models f(h(f(a))) \mapsto g(h(f(a)))$ by Proposition 9 and the claim below.

**Claim 23.** $\mathbf{T}_{\mathcal{P}}\!\downarrow\omega \models h(f(a)) \not\twoheadrightarrow b$.

**Proof.** Any reduction of $h(f(a)) \twoheadrightarrow b$ would have the following form:

$$h(f(a)) \overset{\mathrm{I}}{\twoheadrightarrow} f(z) \overset{\mathrm{II}}{\mapsto} g(z) \overset{\mathrm{III}}{\twoheadrightarrow} b.$$

We will show that II is not a rewrite in $\mathbf{T}_{\mathcal{P}}\!\downarrow\omega$.

Because the total number of $b$-, $c$- and $g$-symbols cannot decrease during (III), $z$ may not contain one of these symbols, hence (I) uses only the two $h$-rules. Therefore, $z$ consists of an odd number of $f$ and $h$ symbols, applied to $a$, so $z \twoheadrightarrow f^{2m+1}(a)$ for some $m$. Using Claim 21 above we get a reduction

$$\mathbf{T}_{\mathcal{P}}\!\uparrow(m+1) \models z \twoheadrightarrow f^{2m+1}(a) \mapsto g(f^{2m}(a)) \twoheadrightarrow b.$$

Then $\mathbf{T}_{\mathcal{P}}\!\downarrow(m+1) \not\models f(z) \mapsto g(z)$. So step II above is indeed absent in $\mathbf{T}_{\mathcal{P}}\!\downarrow\omega$, because this is defined as the intersection of all the $\mathbf{T}_{\mathcal{P}}\!\downarrow m$. □

One might have the idea to reduce the number of rules in the previous example, by identifying $f$, $g$ and $h$. In this way, one more or less gets Example 5. We showed that this system has a unique sound and complete rewrite set, hence it is meaningful. However, contrary to the examples before, for this system the least and greatest fixed points do not coincide. This solves the third open question.

**Example 24.** Let $\mathcal{P}$ be the PRS of Example 5. We have

$$\mathbf{T}_{\mathcal{P}}\!\uparrow 0 = \emptyset$$

$$\mathbf{T}_{\mathcal{P}}\!\downarrow 0 = \{f(a) \mapsto f^3(a)\} \cup \{f^{n+2}(a) \mapsto a \mid n \in \mathbb{N}\}$$

$$\mathbf{T}_{\mathcal{P}}\!\uparrow 1 = \{f(a) \mapsto f^3(a)\}$$

$$\mathbf{T}_{\mathcal{P}}\!\downarrow 1 = \mathbf{T}_{\mathcal{P}}\!\downarrow 0$$

$$\mathbf{T}_{\mathcal{P}}\!\uparrow 2 = \mathbf{T}_{\mathcal{P}}\!\uparrow 1$$

The crux of this system is that, although $\mathbf{T}_{\mathscr{P}} \downarrow 0 \not\models f(a) \mapsto a$, the reduction $f(a) \mapsto f^3(a) \mapsto a$ is still present. Therefore, every closed term reduces to $a$ in $\mathbf{T}_{\mathscr{P}} \downarrow 0$. Clearly, the closure ordinal of this system is 1, but the least and greatest fixed points are not equal.

## 3. Transition system specifications

Not every PRS is meaningful in the sense of Definition 4 (for an example see the appendix). The reason is that a rewrite $f(r) \mapsto s$ is enabled if a certain reduction $r \twoheadrightarrow t$ is *not* present. However, one of these steps may involve the original question, whether $f(r) \mapsto s$ is enabled or not. In [1] this problem is solved by asking for a unique sound and complete rewrite set. A fixed point construction was given to compute the semantics. We showed (Example 24) that this is not a complete method. For some meaningful PRSs the meaning cannot be obtained by this fixed point construction.

In this section, we put the priority mechanism in a wider context. We will present a translation from PRSs into transition system specifications (TSSs). This opens the way to use existing work on operational semantics of TSSs with negative premises [2, 7]. It will turn out (Section 4) that the PRS-semantics coincides with the operational semantics of the TSSs obtained by our translation. In this way, the PRS-semantics gets a broader basis. The translation shows, that the discrepancy between "meaningful" and the fixed point construction is quite inevitable.

A second advantage of the semantics in terms of transition systems is that it provides a way to give semantics to the combination of rule priorities and rules with positive and negative conditions. The conditional rules can be translated to TSS rules in an obvious way. This gives both mechanisms a common basis.

A PRS can be translated to a TSS in a smooth and intuitive way. The addition rules from Example 3 can be translated into:

$$\frac{}{x + 0 \mapsto x} \qquad \frac{y \not\twoheadrightarrow 0}{x + y \mapsto S(x + P(y))}$$

This is not the complete specification, because we also need rules for the context- and transitive closure of $\mapsto$ . (See Definition 33). Although this example illustrates the main idea, it simplifies matters too much. The following example is more representative:

$$\boxed{\begin{array}{c|l} & \mathit{Zero?}(S(y)) \mapsto F \\ \downarrow & \quad \mathit{Zero?}(x) \mapsto T \end{array}}$$

These rules translate into

$$\frac{}{\mathit{Zero?}(S(y)) \mapsto F} \qquad \frac{\forall y. x \not\twoheadrightarrow S(y)}{\mathit{Zero?}(x) \mapsto T}$$

The second rule contains a universal quantifier in the premise. The second rule is enabled if there is no $y$, such that $x$ reduces to $S(y)$. This falls out of the scope of the usual format for negative literals in TSS-theory.

In Section 3.1, we recapitulate some TSS-theory, taken from [7]. On the fly, the format for negative literals will be generalized slightly. In Section 3.2 the translation of priorities into negative premises will be given. In Section 4 the connection with the PRS-semantics is established.

### 3.1. Universal negative premises in TSSs

We assume a signature $\Sigma$ of the form $(\mathscr{F},\mathscr{L},\mathscr{V})$. Here $\mathscr{F}$ is a set of function symbols with fixed arities, $\mathscr{V}$ is an infinite set of variables. Sets of (open) terms $\mathscr{T}(\mathscr{F},\mathscr{V})$ and closed terms $\mathscr{T}(\mathscr{F})$ are defined as usual. Var$(s)$ denotes the variables occurring in term $s$. Furthermore, $\mathscr{L}$ is a set of relation symbols. These occur as names of transitions.

**Definition 25** (*Literals and Rules*). (1) A *positive literal* is of the form $s \longrightarrow^a t$. *Negative literals* are of the form $\forall \vec{z}. s \not\longrightarrow^a t$. Here $s,t \in \mathscr{T}(\mathscr{F},\mathscr{V})$, $\longrightarrow^a \in \mathscr{L}$ and $\vec{z} = \text{Var}(t) - \text{Var}(s) \subset \mathscr{V}$. A literal is *closed* if it contains no free variables (the $\vec{z}$ are not considered free). We let $K$ and $L$ range over arbitrary literals. $H$ and $J$ denote sets of literals; $N$ is reserved for sets of negative literals.

(2) Closed literals $s \longrightarrow^a t$ and $\forall \vec{z}. s \not\longrightarrow^a r$ *deny each other*, if there is a substitution $\sigma$ with $\text{dom}(\sigma) = \vec{z}$, such that $r\sigma = t$. We write $K \lightning L$ if $K$ and $L$ deny each other. Moreover, $H \lightning J$ means that a literal from $H$ denies one from $J$.

(3) A *rule* is of the form $\frac{H}{L}$, with $L$ a positive literal (the conclusion) and $H$ a set of literals (the premises). We often write $L$ for $\frac{\emptyset}{L}$.

(4) A *transition system specification* (TSS) is a set of rules.

The form of negative literals has been generalized in order to capture priorities. We can now dispose of negative literals of the form $s \not\longrightarrow^a$, because they are subsumed by $\forall z. s \not\longrightarrow^a z$. Because $\forall z. s \not\longrightarrow^a t$ can be thought of as an infinite number of ordinary negative premises, the theorems of [7] still apply.

Literals, rules and TSSs will be interpreted by transition relations. These are defined as sets of triples, but can alternatively be seen as families of binary relations (for each relation symbol a relation). It is defined below, when a transition relation is a model for a TSS. Of course, we can only speak about the meaning of a TSS, if there is a way to choose between different models. In case a TSS has only positive rules, the least model is a very natural choice. This model only contains the transitions that are really forced by the rules. This notion is formalized below as positive provability.

**Definition 26** (*Models*). (1) A *transition relation* $R$ is a set of triples of the form $s \longrightarrow^a t$, where $s$ and $t$ are closed terms, and $\longrightarrow^a$ is a relation symbol.

(2) A positive closed literal $L$ *holds in* $R$, if $L \in R$. A negative closed literal $L$ holds in $R$, if there is no closed $K$ that holds in $R$ and such that $K \lightning L$. An open literal holds,

if all its closed instances hold. A set of literals $H$ holds in $R$ if each literal in $H$ holds in $R$.

(3) A rule $\frac{H}{L}$ holds in $R$ if for each closed instance of $H$ that holds in $R$, the corresponding instance of $L$ holds in $R$ too.

(4) A transition relation $R$ is a *model* of a TSS $T$, if each rule from $T$ holds in $R$.

(5) We write $R \models L$ if $L$ holds in $R$. Similarly for rules and sets of literals or rules.

**Definition 27** (*Positive provability*). Given a TSS $T$, *positive provability* (written $\vdash_{+}^{T}$ or $\vdash_{+}$ for short) is inductively defined by the following two clauses:
(1) For any literal $L$, $\{L\} \vdash_{+} L$.
(2) If $\frac{H}{L}$ is an instance of a rule from $T$, and for all $K \in H$, $H_K \vdash_{+} K$, then $\bigcup_{K \in H} H_K \vdash_{+} L$.
We write $\vdash_{+}^{T} L$ for $\emptyset \vdash_{+}^{T} L$.

The following "deduction lemma" and "soundness lemma" will be useful in the sequel.

**Lemma 28.** *If $H \vdash_{+} L$ and for all $K \in H$, $H_K \vdash_{+} K$, then $\bigcup_{K \in H} H_K \vdash_{+} L$.*

**Proof.** Induction on the proof of $L$ from $H$. □

**Lemma 29.** *If $R$ is a model for TSS $T$, and $H \vdash_{+}^{T} L$, then $R \models \frac{H}{L}$.*

**Proof.** Induction over the proof of $L$ from $H$. □

If a TSS $T$ contains positive premises only, it can be viewed as a (simultaneous) inductive definition of a certain labeled transition relation. The relation contains exactly those literals that are provable from $T$. If $T$ contains negative premises in addition, it is not so clear which transition relation is defined. A TSS may even be refused, because it is meaningless. In the full version of [7] up to 11 different solutions for this problem are summarized and compared. Two of these are important for our purpose.

The first one gives a minimality criterion that transition relations should satisfy. The intuition is that positive literals are true only if they are forced somehow. A negative literal may be assumed true, as soon as this is consistent. This intuition is made formal by the notion *well supported model.* A TSS is meaningful, if there is a unique well supported model.

The other method has a proof theoretic flavor. The definition of *positive proof* is extended with a proof rule for deriving *negative* literals. In the second approach, a TSS is meaningful, if each positive literal is either provable or refutable. Unfortunately, these solutions do not coincide.

**Definition 30** (*cf.* [7, Definition 7]; *Well supported transition relations*). A transition relation $R$ is *well supported*[1] by a TSS $T$, if for each positive closed $L$ with $R \models L$, there is a set $N$ of negative literals, such that $N \vdash_{+}^{T} L$ and $R \models N$.

In [7, Proposition 3], it is proved that $T$ has a unique well supported model if and only if it has a least well supported model. In case this exists, it can serve as the semantics of $T$.

We now recapitulate the second method, which adds a new proof rule in order to derive negative information. We dropped the possibility to start with assumptions, because this is not needed. For technical reasons, provability is restricted to closed literals.

**Definition 31** (*cf.* [7, Definition 9]; *Well supported proof*). Given a TSS $T$, *well supported provability* ($\vdash_{\mathrm{ws}}^{T}$ or $\vdash_{\mathrm{ws}}$ for short) is defined inductively by the following two clauses:

(1) If $\frac{H}{L}$ is a closed instance of a rule from $T$, and for all $K \in H$, $\vdash_{\mathrm{ws}} K$, then $\vdash_{\mathrm{ws}} L$.

(2) Let $L$ be a negative closed literal. If for any $K \lightning L$ and set of negative closed literals $N$ such that $N \vdash_{+} K$ we can find an $M$ such that $M \lightning N$ and $\vdash_{\mathrm{ws}} M$, then $\vdash_{\mathrm{ws}} L$.

A TSS $T$ is *complete*, if for each closed transition $s \rightarrow^{a} t$, either $\vdash_{\mathrm{ws}}^{T} s \rightarrow^{a} t$ or $\vdash_{\mathrm{ws}}^{T} s \not\rightarrow^{a} t$.

The second rule has a "negation as failure" flavor: if every attempt to prove a denial $K$ of $L$ fails (because it needs hypotheses $N$ that are in conflict with some $M$ that has been proved already), $L$ may be considered valid. Note that in case no rule matches a transition $s \mapsto t$, then the condition of the second clause is vacuously true, so $\vdash_{\mathrm{ws}} s \not\mapsto t$ holds.

**Proposition 32** (van Glabbeek [7, Proposition 6]). *Let $T$ be a TSS.*

(1) $\vdash_{\mathrm{ws}}^{T}$ *is consistent.*

(2) *If $\vdash_{\mathrm{ws}}^{T} L$ then $R \models L$ for all well supported models $R$ of $T$.*

### 3.2. Translation of PRSs into TSSs

In this section, we give a translation of a PRS $\mathscr{P}$ to TSS($\mathscr{P}$). Without loss of generality, we make two assumptions about $\mathscr{P}$. The first is that different rules have disjoint variables. This can always be reached by renaming the variables. The second assumption is, that for each inhabited arity $m$ in $\mathscr{P}$, there is an $\dot{m}$-ary function symbol that does not occur in the rules, denoted by $\langle \_, \ldots, \_ \rangle_{m}$. This can always be achieved by adding new function symbols. This is to avoid $\twoheadrightarrow^{\mathrm{int}}$ as a relation symbol.

[1] This use of the phrase "well supported" coincides with the full version of [7]. In the extended abstract [7], the same notion is still used to define stability. But the phrase "well supported" is used there for a more complex notion, and it is stated that the stable models and the well supported models coincide.

**Definition 33** (*The translation*). (1) Let $\Sigma=(\mathscr{F},\mathscr{V})$. Put $\mathsf{TSS}(\Sigma)=(\mathscr{F},\mathscr{V},\mathscr{L})$, where $\mathscr{L}:=\{\mapsto,\rightarrow,\twoheadrightarrow\}$.

(2) Let $\mathscr{P}=(\mathscr{R},>)$ be given. Let $x=f(\vec{s})\mapsto t$ be a rule in $\mathscr{R}$. Define

$$\mathsf{TSS}(x)=\frac{\{\forall\vec{z}.\,\langle\vec{s}\rangle\not\twoheadrightarrow\langle\vec{r}\rangle\mid(f(\vec{r})\mapsto t')>x\text{ in }\mathscr{P},\text{ and }\vec{z}=\mathrm{Var}(\vec{r})\}}{x}$$

(3) Depending on $\Sigma$, a set of rules $F$ is defined, consisting of

$$\frac{x_i\rightarrow y}{f(x_1,\ldots,x_i,\ldots,x_n)\rightarrow f(x_1,\ldots,y,\ldots,x_n)}\ (\mathrm{C2})$$

$$\frac{x\mapsto y}{x\rightarrow y}\ (\mathrm{C1})\qquad\frac{}{x\twoheadrightarrow x}\ (\mathrm{T1})\qquad\frac{x\rightarrow y\quad y\twoheadrightarrow z}{x\twoheadrightarrow z}\ (\mathrm{T2})$$

Rule C2 is present for each function symbol $f$, including $\langle\_\rangle_m$, and for each $1\leqslant i\leqslant \mathsf{arity}(f)$ (so not for constants).

(4) Let $\mathscr{P}=(\mathscr{R},>)$ with $\mathscr{R}=(\Sigma,R)$ be given. Define $\mathsf{TSS}(\mathscr{P})=(\mathsf{TSS}(\Sigma),R')$, where $R'=\{\mathsf{TSS}(r)\mid r\in R\}\cup F$.

**Example 34.** Let $\mathscr{P}$ be the following PRS [1, Example 4]:

$$\boxed{\begin{array}{l} Eq(x,x)\mapsto T \\ \downarrow Eq(x,y)\mapsto F \end{array}}$$

The TSS associated to $\mathscr{P}$ has the following rules:

$$\frac{}{Eq(x,x)\mapsto T}\qquad\frac{\forall z.\,\langle x,y\rangle\not\twoheadrightarrow\langle z,z\rangle}{Eq(x,y)\mapsto F}\qquad\frac{x\mapsto y}{x\rightarrow y}$$

$$\frac{x\rightarrow y}{Eq(x,z)\rightarrow Eq(y,z)}\qquad\frac{x\rightarrow y}{Eq(z,x)\rightarrow Eq(z,y)}\qquad\frac{x\rightarrow y}{\langle x,z\rangle\rightarrow\langle y,z\rangle}$$

$$\frac{x\rightarrow y}{\langle z,x\rangle\rightarrow\langle z,y\rangle}\qquad\frac{}{x\twoheadrightarrow x}\qquad\frac{x\rightarrow y\quad y\twoheadrightarrow z}{x\twoheadrightarrow z}$$

Any transition relation for $\mathsf{TSS}(\mathscr{P})$ can be seen as a triple of binary relations $(R,C,T)$, where $R$ interprets $\mapsto$, $C$ interprets $\rightarrow$ and $T$ interprets $\twoheadrightarrow$. Any rewrite set $R$ gives rise to the transition relation $(R,\rightarrow_R,\twoheadrightarrow_R)$. We use $R\models L$ as an abbreviation of $(R,\rightarrow_R,\twoheadrightarrow_R)\models L$. Note that if $(R,C,T)$ is an arbitrary transition relation for $\mathsf{TSS}(\mathscr{P})$, then $R$ is not necessarily a rewrite set for $\mathscr{P}$ (i.e. a set of closed rule instances), nor is it always the case that $C=\rightarrow_R$ and $T=\twoheadrightarrow_R$.

The adequacy of the translation above is shown by the following lemma, which is also the key lemma in subsequent sections.

**Lemma 35.** *Let $\mathscr{P}$ be a PRS, $R$ a set of $\mathscr{P}$-rewrites. Put $T := \mathsf{TSS}(\mathscr{P})$ and let $L$ be a positive closed literal of $T$. Then*

$$R^{\mathscr{P}} \models L \Leftrightarrow \textit{ for some set } N \textit{ of negative premises, } N \vdash_{+}^{T} L \textit{ and } R \models N$$

**Proof.** $\Rightarrow$: Distinguish the three possible forms of $L$.

(1) $L = s \longmapsto t$ with $s = f(\vec{s})$, an instance of rule $x$. Put

$$N := \{\forall\vec{z}.\langle\vec{s}\rangle \not\twoheadrightarrow \langle\vec{a}\rangle \mid (f(\vec{a}) \longmapsto b) > x \text{ in } \mathscr{P} \text{ and } \mathrm{Var}(\vec{a}) = \vec{z}\}.$$

Then $\frac{N}{L}$ is an instance of the rule $\mathsf{TSS}(x)$. Clearly $N \vdash_{+} L$. Furthermore, $L$ is in $R^{\mathscr{P}}$, so it is correct w.r.t. $R$. So for any $\sigma$-instance of any rule $(f(\vec{a}) \longmapsto b) > x$, we have the following: $f(\vec{s}) \not\twoheadrightarrow_{R}^{\mathrm{int}} f(\vec{a})^{\sigma}$. So $\langle\vec{s}\rangle \not\twoheadrightarrow_{R} \langle\vec{a}\rangle^{\sigma}$. In other words, $R \models N$.

(2) $L = s \longrightarrow t$. Then for some $C[\,]$, $l$ and $r$, $s = C[l]$, $t = C[r]$ and $R^{\mathscr{P}} \models l \longmapsto r$. By (1), there is a set $N$ such that $R \models N$ and $N \vdash_{+} l \longmapsto r$. By C1, $N \vdash_{+} l \longrightarrow r$. By suitable applications of C2, $N \vdash_{+} s \longrightarrow t$.

(3) $L = s \twoheadrightarrow t$. Then for some $n$, $s_0, \ldots, s_n$, we have $s = s_0$, $t = s_n$, and for all $0 \leqslant i < n$, $s_i \longrightarrow s_{i+1}$. By (2), there are $N_i$ such that $R \models N_i$ and $N_i \vdash_{+} s_i \longrightarrow s_{i+1}$. Put $N := \bigcup N_i$. Then $R \models N$, and by suitable applications of T1 and T2, also $N \vdash_{+} s \twoheadrightarrow t$.

$\Leftarrow$: Induction on $N \vdash_{+} L$. We distinguish the last applied rule in the proof:

(1) Application of $\mathsf{TSS}(x)$ for some rule $x$. This is the only step of the proof, because negative premises can only occur as assumptions in positive proofs. Therefore, $\mathsf{TSS}(x)$ is $\frac{N}{L}$. $R \models N$, hence $L$ is correct w.r.t. $R$, so $R^{\mathscr{P}} \models L$. (Details are similar as in $\Rightarrow$).

(2) Application of C1. Then $L$ is of the form $s \longrightarrow t$, and $N \vdash_{+} s \longmapsto t$ is a subproof. By induction hypothesis, $R^{\mathscr{P}} \models s \longmapsto t$, hence also $R^{\mathscr{P}} \models s \longrightarrow t$.

(3) Application of C2. Then $L$ is of the form $f(\ldots,s,\ldots) \longrightarrow f(\ldots,t,\ldots)$, and $N \vdash_{+} s \longrightarrow t$ is a subproof. By induction hypothesis, $R^{\mathscr{P}} \models s \longrightarrow t$, hence also $R^{\mathscr{P}} \models f(\ldots,s,\ldots) \longrightarrow f(\ldots,t,\ldots)$.

(4) Application of T1. Then $L$ is of the form $s \twoheadrightarrow s$. Clearly, $R^{\mathscr{P}} \models s \twoheadrightarrow s$.

(5) Application of T2. Then $L$ is of the form $s \twoheadrightarrow t$, and the subproofs have the form $N_1 \vdash_{+} s \longrightarrow r$ and $N_2 \vdash_{+} r \twoheadrightarrow t$, for some $r$ and $N_1 \cup N_2 = N$. By induction hypothesis, $R^{\mathscr{P}} \models s \longrightarrow r$ and $R^{\mathscr{P}} \models r \twoheadrightarrow t$, hence also $R^{\mathscr{P}} \models s \twoheadrightarrow t$. $\square$

## 4. Operational semantics of PRSs

We now want to establish a link between the PRS-semantics and the semantics that comes with transition systems. The comparison is made possible by our translation. Indeed, there is a quite remarkable connection. We will show (Theorem 39) that the sound and complete rewrite sets for $\mathscr{P}$ coincide with the well supported models of $\mathsf{TSS}(\mathscr{P})$. To this end, it is proved that a rewrite set is complete for $\mathscr{P}$ if and only if it is a model for $\mathsf{TSS}(\mathscr{P})$. In the same way, soundness and well-supportedness are tightly related.

In Section 4.2, we will also establish a link between complete TSSs and the fixed point construction for PRSs. It will turn out (Theorem 42) that TSS($\mathscr{P}$) is a complete specification if and only if the least and greatest fixed points of the operator $\mathrm{T}_{\mathscr{P}}$ coincide.

### *4.1. Sound and complete vs. well supported model*

Recall that $R$ is complete if it contains all correct rewrites w.r.t. itself. Therefore, a rewrite is present whenever the negative premises connected to it are true. This in turn means that the rules of the associated TSS are true, hence the rewrite set is a model. Hence, a rewrite set is complete for $\mathscr{P}$ if and only if it is a model for TSS($\mathscr{P}$).

**Proposition 36.** *$R$ is a complete rewrite set of a PRS $\mathscr{P}$ if and only if $(R, \to_R, \twoheadrightarrow_R)$ is a model of* TSS($\mathscr{P}$).

**Proof.** $\Rightarrow$: Rules C1, C2, T1 and T2 clearly hold in $(R, \to_R, \twoheadrightarrow_R)$. Now let some other rule, $\frac{N}{r}$ be given. Assume that $R \models N^\sigma$, for some substitution $\sigma$. Then by Lemma 35, $R^{\mathscr{P}} \models r^\sigma$. Because $R$ is complete, also $R \models r^\sigma$ (Lemma 7.2). Now $(R, \to_R, \twoheadrightarrow_R)$ is a model of TSS($\mathscr{P}$), because all rules hold in it.

$\Leftarrow$: Let $s \mapsto^r t$ be correct w.r.t. $R$; then it is in $R^{\mathscr{P}}$. By Lemma 35, there exist negative premises $N$, such that $R \models N$ and $N \vdash_+ s \mapsto^r t$. Because $R$ is a model of TSS($\mathscr{P}$), Lemma 29 yields $R \models s \mapsto t$. Hence $R$ is complete. □

Now we will show that the sound rewrite sets coincide with the well supported models. The intuition is that in a sound rewrite set, all rewrites are correct, so the negative premises connected with them are true. The latter forms the basic idea of well-supportedness.

**Proposition 37.** *Let $\mathscr{P}$ be a PRS. Then $R$ is a sound rewrite set if and only if $(R, \to_R, \twoheadrightarrow_R)$ is a well supported transition relation for* TSS($\mathscr{P}$).

**Proof.** $\Rightarrow$: Let $R \models L$ for some $L$. As $R$ is sound, all redexes used in the reduction $L$ are correct w.r.t. $R$, hence also $R^{\mathscr{P}} \models L$. By Lemma 35 there exists a set $N$ of negative premises, such that $R \models N$ and $N \vdash_+ L$. Hence $(R, \to_R, \twoheadrightarrow_R)$ is well supported by TSS($\mathscr{P}$).

$\Rightarrow$: Assume $R \models s \mapsto t$. By well-supportedness, there is a set $N$ of negative premises such that $R \models N$ and $N \vdash_+ s \mapsto t$. By Lemma 35, $R^{\mathscr{P}} \models s \mapsto t$, so $s \mapsto t$ is a correct rewrite w.r.t. $R$. Hence $R$ is sound. □

Together, Propositions 36 and 37 show that sound and complete rewrite relations coincide with well supported models of the form $(R, \to_R, \twoheadrightarrow_R)$. We still have to show that a well supported model has this particular form.

**Lemma 38.** *Let a PRS $\mathscr{P}$ be given. Any well supported model of* TSS($\mathscr{P}$) *is of the form* $(R, \rightarrow_R, \twoheadrightarrow_R)$ *for some rewrite set R.*

**Proof.** Let $(R, C, T)$ be a well supported model. Because it is a model of C1, C2, T1 and T2, $\rightarrow_R \subseteq C$ and $\twoheadrightarrow_R \subseteq T$.

Next, let $(R, C, T) \models L$, for arbitrary positive $L$. Then by well-supportedness, for some set $N$ of negative premises, $(R, C, T) \models N$ and $N \vdash^T_+ L$. By induction on this proof it can be shown that if $L$ is of the form $s \mapsto t$ then it is a rewrite; if $L$ is of the form $s \rightarrow t$ then $s \rightarrow_R t$ and if $L$ is of the form $s \twoheadrightarrow t$ then $s \twoheadrightarrow_R t$. (details are similar to the proof of Lemma 35). □

For the previous lemma, we really need that the transition relation is *well* supported. There exists a less restrictive notion of supportedness, but in Appendix A we give an example showing that this is not enough.

We are now able to state the main theorem of this section. The theorem says that the PRS-semantics can be expressed in terms of models of TSSs.

**Theorem 39.** *Let $\mathscr{P}$ be a PRS. The following two statements are equivalent:*
(1) *$\mathscr{P}$ has a unique sound and complete rewrite set.*
(2) TSS($\mathscr{P}$) *has a least well supported model.*

**Proof.** Any sound and complete rewrite set $R$ for $\mathscr{P}$ yields a well supported model $(R, \rightarrow_R, \twoheadrightarrow_R)$ for TSS($\mathscr{P}$), by Lemmas 36 and 37. Conversely, each well supported model is of the form $(R, \rightarrow_R, \twoheadrightarrow_R)$, where $R$ is a sound and complete rewrite set, by Lemmas 38, 36 and 37. By [7, Proposition 3], if a least well supported model exists, then this is the unique well supported model. Now the theorem follows. □

### *4.2. Fixed points and complete specifications*

Recall from Section 2.2 the function $(\,)^{\mathscr{P}}$, which assigns to each rewrite set $R$ the set of correct rewrites. We had a series $\mathbf{T}_{\mathscr{P}}\uparrow\alpha$ iterating $(\,)^{\mathscr{P}}$ an even number of times, starting with $\emptyset$, and $\mathbf{T}_{\mathscr{P}}\downarrow\alpha$, iterating $(\,)^{\mathscr{P}}$ odd times, starting with $\emptyset^{\mathscr{P}}$. In this way we obtained the least and greatest fixed points of $\mathbf{T}_{\mathscr{P}}$ that, when equal, yield the unique fixed point of $(\,)^{\mathscr{P}}$.

This section is devoted to the proof that these fixed points coincide if and only if TSS($\mathscr{P}$) is a complete transition system specification. We have to relate truth in $\mathbf{T}_{\mathscr{P}}\uparrow\alpha$ and $\mathbf{T}_{\mathscr{P}}\downarrow\alpha$ with provability of positive and negative literals. In Proposition 40 we show that for any $\alpha$, $\mathbf{T}_{\mathscr{P}}\uparrow\alpha$ only contains information that is provable. This is proved simultaneously with the fact that only refutable transitions are outside $\mathbf{T}_{\mathscr{P}}\downarrow\alpha$.

**Proposition 40.** *Let PRS $\mathscr{P}$ with* TSS($\mathscr{P}$) $= T$ *and ordinal $\alpha$ be given. Then we have:*
(1) *For all positive closed $L$, if $\mathbf{T}_{\mathscr{P}}\uparrow\alpha \models L$ then $\vdash^T_{\mathrm{ws}} L$.*
(2) *For all negative closed $L$, if $\mathbf{T}_{\mathscr{P}}\downarrow\alpha \models L$ then $\vdash^T_{\mathrm{ws}} L$.*

**Proof.** Simultaneous induction on $\alpha$. We first prove that for fixed $\alpha$, we have $(1) \Rightarrow (2)$. Let $\alpha$ be fixed, assume (1) and $\mathbf{T}_{\mathscr{P}}\downarrow\alpha \models L$ for arbitrary negative $L$. In order to apply proof rule 31.2, let $K$ and $N$ be given, such that $K \lightning L$ and $N \vdash_+ K$. Then $\mathbf{T}_{\mathscr{P}}\downarrow\alpha \not\models K$. By 9.1 and 35, $\mathbf{T}_{\mathscr{P}}\uparrow\alpha \not\models N$. So there exists some (positive) $M$, with $M \lightning N$ and $\mathbf{T}_{\mathscr{P}}\uparrow\alpha \models M$. By (1), $\vdash^{T}_{\mathrm{ws}} M$. Hence $\vdash^{T}_{\mathrm{ws}} K$.

Next we prove (1) by ordinal induction on $\alpha$. By the implication above, we may also use the induction hypothesis of (2).

0: $\mathbf{T}_{\mathscr{P}}\uparrow 0 = \emptyset$, Then $L$ is of the form $s \twoheadrightarrow s$, which is provable by T1.

$\alpha+1$: Let $\mathbf{T}_{\mathscr{P}}\uparrow(\alpha+1) \models L$. By Lemma 35 and 9.2, there is a set $N$ of negative premises, such that $N \vdash_+ L$ and $\mathbf{T}_{\mathscr{P}}\downarrow\alpha \models N$. By induction hypothesis (2), $\vdash^{T}_{\mathrm{ws}} N$. Hence $\vdash^{T}_{\mathrm{ws}} L$.

$\lambda$ (a limit ordinal): Let $\mathbf{T}_{\mathscr{P}}\uparrow\lambda \models L$. Let $I$ be the set of redexes used in the reduction $L$. $\mathbf{T}_{\mathscr{P}}\uparrow\lambda \models I$. Because $I$ is finite, there is some $\alpha < \lambda$, such that $\mathbf{T}_{\mathscr{P}}\uparrow\alpha \models I$. Then also $\mathbf{T}_{\mathscr{P}}\uparrow\alpha \models L$, and by induction hypothesis (1), $\vdash^{T}_{\mathrm{ws}} L$. □

The next proposition serves as the converse of the previous one. It expresses that the provable transitions hold in fixed points of $\mathbf{T}_{\mathscr{P}}$ and that refutable transitions are not correct w.r.t. them.

**Proposition 41.** *Let PRS $\mathscr{P}$ with* $\mathsf{TSS}(\mathscr{P}) = T$ *be given. Let a rewrite set $R$ be given, with $(R^{\mathscr{P}})^{\mathscr{P}} = R$. Let $L$ be a closed literal.*

(1) *If $\vdash^{T}_{\mathrm{ws}} L$ and $L$ is positive then $R \models L$.*

(2) *If $\vdash^{T}_{\mathrm{ws}} L$ and $L$ is negative then $R^{\mathscr{P}} \models L$.*

**Proof.** Simultaneous induction on the definition of $\vdash_{\mathrm{ws}}$. Distinguish the last step in this proof.

- The last step is $\frac{N}{L}$ for some negative set of premises $N$, and $L = s \mapsto t$. We have smaller subproofs $\vdash^{T}_{\mathrm{ws}} N$ so by induction hypothesis (2), $R^{\mathscr{P}} \models N$. Clearly $N \vdash_+ L$. By Lemma 35, $(R^{\mathscr{P}})^{\mathscr{P}} \models L$. Because $(R^{\mathscr{P}})^{\mathscr{P}} = R$ we have $R \models L$.
- The last step is an application of C1, C2, T1 or T2. These cases follow straightforwardly from the induction hypotheses.
- $L$ is negative and the last step is an application of the rule 31.2. Let $K$ and $N$ be given, such that $K \lightning L$ and $N \vdash_+ K$. Then we have immediate subproofs of $\vdash_{\mathrm{ws}} M$, for some $M$ with $M \lightning N$. By induction hypothesis (1), $R \models M$. Hence $R \not\models N$. So for all $N$ with $N \vdash_+ K$, we have $R \not\models N$. Hence $R^{\mathscr{P}} \not\models K$ by Lemma 35. This holds for any $K$ with $K \lightning L$, hence $R^{\mathscr{P}} \models L$. □

We are now able to prove that $\mathbf{T}_{\mathscr{P}}$ has a unique fixed point if and only if the specification of $\mathsf{TSS}(\mathscr{P})$ is complete.

**Theorem 42.** *Let $\mathscr{P}$ be a PRS, with closure ordinal $\alpha$. Then $\mathbf{T}_{\mathscr{P}}\uparrow\alpha = \mathbf{T}_{\mathscr{P}}\downarrow\alpha$ if and only if* $\mathsf{TSS}(\mathscr{P})$ *is complete.*

**Proof.** $\Rightarrow$: For each positive closed literal $L$, either $\mathbf{T}_{\mathscr{P}}\uparrow\alpha \models L$ or $\mathbf{T}_{\mathscr{P}}\uparrow\alpha \not\models L$. In the first case, $\vdash_{\mathrm{ws}} L$ by Proposition 40(1). Otherwise, $\mathbf{T}_{\mathscr{P}}\downarrow\alpha \not\models L$, as we may assume $\mathbf{T}_{\mathscr{P}}\uparrow\alpha = \mathbf{T}_{\mathscr{P}}\downarrow\alpha$. Hence $\mathbf{T}_{\mathscr{P}}\downarrow\alpha \models \neg L$, and by Proposition 40(2), $\vdash_{\mathrm{ws}} \neg L$.

$\Leftarrow$: Note that for the closure ordinal $\alpha$, $\mathbf{T}_{\mathscr{P}}\uparrow\alpha = ((\mathbf{T}_{\mathscr{P}}\uparrow\alpha)^{\mathscr{P}})^{\mathscr{P}}$, so Proposition 41 is applicable. Note also that $\mathbf{T}_{\mathscr{P}}\uparrow\alpha \subseteq \mathbf{T}_{\mathscr{P}}\downarrow\alpha = (\mathbf{T}_{\mathscr{P}}\uparrow\alpha)^{\mathscr{P}}$. (By Propositions 7(1), 9(1) and 10(1)). We still have to prove $\mathbf{T}_{\mathscr{P}}\uparrow\alpha \supseteq \mathbf{T}_{\mathscr{P}}\downarrow\alpha$. Let $\mathbf{T}_{\mathscr{P}}\downarrow\alpha \models s \mapsto t$, then (Proposition 41(2)) $\not\models_{\mathrm{ws}} s \not\mapsto t$. Hence by completeness of TSS($\mathscr{P}$), $\vdash_{\mathrm{ws}} s \mapsto t$. Now by Proposition 41(1), $\mathbf{T}_{\mathscr{P}}\uparrow\alpha \models s \mapsto t$. This shows that $\mathbf{T}_{\mathscr{P}}\downarrow\alpha \subseteq \mathbf{T}_{\mathscr{P}}\uparrow\alpha$. □

## 5. Conclusion

We summarize the findings of the paper. In Table 1, the counterexamples presented earlier are mentioned, with the properties that they illustrate. Table 2 compares a PRS $\mathscr{P}$ with its translation TSS($\mathscr{P}$) (Definition 33). The numbers refer to the theorems where the correspondence is proved. The third result is that bounded PRSs are executable (Theorem 18).

## Acknowledgements

The research on which this paper is based was carried out at the Philosophy Department of the Utrecht University, The Netherlands. I am grateful to Jan Bergstra, who suggested this topic to me. I thank Wan Fokkink for the fruitful discussions with him and for directing me to Rob van Glabbeek, and Rob for his comments. Finally, I am indebted to the anonymous referees for their useful suggestions.

Table 1
Counterexamples to open problems

| PRS | Illustrated property |
|---|---|
| Example 19 | Closure ordinal $\omega$ |
| Example 22 | Closure ordinal $>\omega$, so $\mathbf{T}_{\mathscr{P}}$ is not continuous |
| Examples 5, 24 | meaningful, but no unique fixed point. |

Table 2
Comparison of notions for PRSs with counterparts for TSSs

| Notion for $\mathscr{P}$ | Notion for TSS($\mathscr{P}$) | Theorem |
|---|---|---|
| Complete rewrite set | Model | Prop. 36 |
| Sound rewrite set | Well supported relation | Prop. 37 |
| Unique soundandcomplete rewrite set | Least well supported model | Thm. 39 |
| Unique fixed point | Complete TSS | Thm. 42 |

## Appendix A. Just supported is not enough

This appendix gives an example that serves as extra explanation. In Section 4.1 we proved that the sound and complete rewrite sets for PRS $\mathscr{P}$ correspond to well supported models of TSS($\mathscr{P}$). The definition of well-supportedness is quite intricate, as it requires that for each transition $L$ in a model $\mathcal{M}$, there exist negative premises $N$, such that $N \vdash_+ L$ and $\mathcal{M} \models N$. There exists a much simpler definition of supportedness:

**Definition 43** (*van Glabbeek* [7, Definition 5]). A transition relation $\mathcal{M}$ is *supported* if for every transition $L \in \mathcal{M}$, there is a rule instance $\frac{H}{L}$ such that $\mathcal{M} \models H$.

Instead of the existence of a proof of $L$ with true negative premises, now simply a rule with conclusion $L$ is required, with true premises. However, note that when $L$ appears among $H$, then the support for $L$ is not very convincing. In this case the presence of $L$ would be used to make sure that $L$ is forced. Such a circularity can also be less visible. The circularity is avoided in the definition of *well*-supportedness (Definition 30). Indeed, we can find an example of a PRS that has no sound and complete rewrite set, but whose corresponding TSS has a least supported model. In the sequel, $\mathscr{P}$ refers to the following example.

**Example 44** (*Baeten et al.* [1, Example 2.12]).

$$\begin{tikzcd}[row sep=small] 1 \mapsto A(1) \\ A(0) \mapsto 1 \arrow[d] \\ A(x) \mapsto 0 \end{tikzcd}$$

In [1] it is shown that this system has no meaning. The problem lies in the fact that the rewrite $A(1) \mapsto 0$ is allowed if and only if $1 \not\twoheadrightarrow 0$. This however is precisely the case if $A(1) \not\mapsto 0$. Hence no sound and complete rewrite set can exist.

Applying the translation of Definition 33, we get a TSS consisting of the fixed rules C1, C2, T1 and T2, together with:

$$\frac{}{1 \mapsto A(1)}\ (\mathrm{R1}) \qquad \frac{}{A(0) \mapsto 1}\ (\mathrm{R2}) \qquad \frac{x \not\twoheadrightarrow 0}{A(x) \mapsto 0}\ (\mathrm{R3})$$

Remember that models of TSS($\mathscr{P}$) are of the form $(R, C, T)$, where $R$ is the rewrite set, and $C$ and $T$ interpret the context- and transitive closure, respectively. As shown below, *supportedness* does not guarantee that $T$ really equals $\twoheadrightarrow_R$. Lemma 38 shows that for *well*-supported models, this is guaranteed.

We claim that $\mathscr{M} := (R, C, T)$ as defined below is the least supported model of $\mathsf{TSS}(\mathscr{P})$. This is proved by (1), (2) and (3) below.

$$R := \{(1, A(1)),\ (A(0), 1)\}$$

$$C := \rightarrow_R$$

$$T := \ \twoheadrightarrow_R \cup \{(x, 0) \mid x \text{ a closed term}\}$$

We have no R3-rewrites in $R$. So in order to make R3 true, its premise must be false. This is done by ensuring that each term "reduces" to 0 in $T$. We have

(1) *$\mathscr{M}$ is a model of* $\mathsf{TSS}(\mathscr{P})$. Clearly, R1, R2, C1, C2 and T1 hold in $\mathscr{M}$. R3 holds, because its premise is never true. As to T2, assume $x \rightarrow_R y$ and $yTz$. Now either $y \twoheadrightarrow_R z$, in which case also $x \twoheadrightarrow_R z$, or $z = 0$. In both cases we have $xTz$. Hence T2 also holds in $\mathscr{M}$.

(2) *$\mathscr{M}$ is supported.* Elements of $R$ are supported by rules R1 or R2. Elements of $C$ are supported by rules C1 or C2. The $\twoheadrightarrow_R$-elements of $T$ are supported by T1 or T2. Finally, the $(x, 0)$ elements of $T$ can be supported as follows. If $x = 0$, then T1 supports it. For $x = A^n(1)$, we find as support

$$\frac{A^n(1) \rightarrow A^{n+1}(1) \quad A^{n+1}(1) \twoheadrightarrow 0}{A^n(1) \twoheadrightarrow 0}\ (\mathrm{T2})$$

Both premises hold in $\mathscr{M}$. For $x = A^{n+1}(0)$ we find as support

$$\frac{A^{n+1}(0) \rightarrow A^n(1) \quad A^n(1) \twoheadrightarrow 0}{A^{n+1}(0) \twoheadrightarrow 0}\ (\mathrm{T2})$$

Again, both premises are true in $\mathscr{M}$.

(3) *$\mathscr{M}$ is contained in any supported model $\mathscr{M}' := (R', C', T')$.* As $\mathscr{M}'$ is a model of $\mathsf{TSS}(\mathscr{P})$, surely $R \subseteq R'$ (because R1 and R2 hold); from C1 and C2 we derive $C = \rightarrow_R \subseteq \rightarrow_{\mathscr{M}'} \subseteq C'$; and by T1 and T2, we have $\twoheadrightarrow_R \subseteq C'^* \subseteq T'$.

We still have to show that $(x, 0) \in T'$ for all closed $x$. If $x$ is 0, this follows by T1. Assume towards a contradiction, that $T' \models A^n(1) \not\twoheadrightarrow 0$. R3 holds in $\mathscr{M}'$, so $R' \models A^{n+1}(1) \mapsto 0$, hence by C1, T1, T2, $T' \models A^{n+1}(1) \twoheadrightarrow 0$. Using R1, C1, C2, we derive $C' \models A^n(1) \rightarrow A^{n+1}(1)$. Now by T2, we get $T' \models A^n(1) \twoheadrightarrow 0$. Contradiction. Hence $T' \models A^n(1) \twoheadrightarrow 0$. But then, using R2, C1, C2, T2, we also obtain $T' \models A^{n+1}(0) \twoheadrightarrow 0$ (via $A^n(1)$). Hence, $T \subseteq T'$.

(1), (2) and (3) together yield that $(R, C, T)$ is the least supported model of $\mathsf{TSS}(\mathscr{P})$. This shows that Theorem 39 is not true if we replace "well supported" by "supported".

## References

[1] J.C.M. Baeten, J.A. Bergstra, J.W. Klop, W.P. Weijland, Term-rewriting systems with rule priorities, Theoret. Comput. Sci. 67 (2–3) (1989) 283–302.

[2] R. Bol, J.F. Groote, The meaning of negative premises in transition system specifications, J. ACM 43 (5) (1996) 863–914.

[3] W.J. Fokkink, C. Verhoef, A conservative look at term deduction systems with variable binding, Logic Group Preprint Series 140, Utrecht University, 1995.

[4] J.R. Kennaway, The specificity rule for lazy pattern-matching in ambiguous term rewrite systems, in: Proc. 3rd European Symp. on Programming, Lecture Notes in Computer Science, vol. 432, Springer, Berlin, 1990, pp. 256–270.

[5] A. Laville, Comparison of priority rules in pattern matching and term rewriting, J. Symbol. Comput. 11 (1991) 321–347.

[6] C.K. Mohan, Priority rewriting: semantics, confluence, and conditionals, in: N. Dershowitz (Ed.), Proc. 3rd Internat. Conf. on Rewriting Techniques and Applications, Chapel Hill, NC, April 1989. Lecture Notes in Computer Science, vol. 355, Springer, Berlin, 1989, pp. 278–291.

[7] R.J. van Glabbeek, The meaning of negative premises in transition system specifications II (extended abstract), in: F. Meyer auf der Heide, B. Monien (Eds.), Automata, Languages and Programming, 23rd ICALP, Lecture Notes in Computer Science, vol. 1099, Springer, Berlin, 1996, pp. 502–513. (Full version appeared as technical note STAN-CS-TN-95-16).