International Journal of Foundations of Computer Science

TERMINATION OF ABSTRACT REDUCTION SYSTEMS

JEREMY E DAWSON
Logic and Computation Program, NICTA, Locked Bag 8001, Canberra ACT 2601, AUSTRALIA,
RSISE (Bldg 115) Australian National University Canberra ACT 0200 AUSTRALIA

and

RAJEEV GORÉ
RSISE (Bldg 115) Australian National University Canberra ACT 0200 AUSTRALIA

Received (received date)
Revised (revised date)
Communicated by Editor's name

ABSTRACT

We present a general theorem capturing conditions required for the termination of abstract reduction systems. We show that our theorem generalises another similar general theorem about termination of such systems. We apply our theorem to give interesting proofs of termination for typed combinatory logic. Thus, our method can handle most path-orderings in the literature as well as the reducibility method typically used for typed combinators. Finally we show how our theorem can be used to prove termination for incrementally defined rewrite systems, including an incremental general path ordering. All proofs have been formally machine-checked in Isabelle/HOL.

Keywords: rewriting, termination, well-founded ordering, strong normalisation

WE PROPOSE TO OMIT THE TWO EXAMPLES INDICATED LATER, UNLESS THE EDITORS/REFEREES HAVE DIFFERENT SUGGESTIONS. THIS WILL BRING IT DOWN WITHIN 25 PAGES

1. Introduction

We address the general problem of termination of rewriting which can be informally posed as follows. Assume that we have a fixed set of "objects" defined according to some formal syntax. Suppose we are given a binary relation ρ on these objects, where $(t, s) \in \rho$ expresses that object s may be transformed into object t. Let us call such a transformation a *reduction*. Consider repeatedly reducing an object in any way possible. We are interested to know whether such repeated reduction necesarily terminates – formally, whether or not there is an infinite sequence $t = t_0, t_1, \ldots$, where each $(t_{i+1}, t_i) \in \rho$. If there is not, we say that t is strongly normalising: $t \in SN$. The difficulty of the problem arises from the totally general notion of "reduction".

In the common special case of a term rewriting system (TRS), an object is a term of some first-order language, and the reduction relation is described by a set of "rewrite rules" $l_i \to r_i$, where l_i and r_i are terms containing variables for which terms may be substituted. Here, rewriting is also usually *monotonic* [5], or *closed under context* [20], in that if $C[\_]$, a term with a "hole", is the context, and l reduces to r, then $C[l]$ reduces to $C[r]$. There are several general methods capturing termination of such term rewriting systems [3, 14, 8].

Recently, Jean Goubault-Larrecq [19] has proved termination results for the more general setting of an abstract reduction system (ARS) where objects need not have a term structure and so there is no monotonicity assumption or subterm relation. Where proofs for TRSs involve the subterm relation, he uses an arbitrary well-founded relation $\lhd$ in a similar way. His Theorem 1, let us call it JGL1, is the result of "chasing generalizations and simplifications" of earlier work and "subsumes ... most path-orderings of the literature" [19].

Among the results of this paper, the first we proved was Theorem 7, as [8, Theorem 2]. While this theorem and JGL1 use remarkably similar ideas, it was frustrating to see that neither JGL1 nor our Theorem 7 actually subsumed the other, even though [8] applies to TRSs and [19] applies to ARSs. Indeed, there is something in common between the two results: a more general theorem that subsumes both.

We first present this more general theorem and its proof in §2. Then, in §3, we show that it generalises JGL1. In §4, we show how to use our new theorem to obtain two different proofs of strong normalisation for well-typed combinator terms. In these cases, although the objects have a term structure, our proofs use the ARS setting and show the well-foundedness of larger, non-monotonic, relations. The combinator case suggests that our theorem properly generalises JGL1. Thus, our theorem handles most path-orderings in the literature and the reducibility method typically used for typed combinators.

In §3.2 we show directly how our result implies the termination of the general path ordering of Dershowitz & Hoot [14].

In §5 we show how the main theorem implies a similar result for TRSs [8, Theorem 2], and we relate that result to work on constricting derivations.

We then consider applications of the theorem for TRSs, showing how it gives well-known results for the Knuth-Bendix Ordering, and relating to dependency pairs. Also, by an analogue of §3.2 (see [8, §3.7]), the termination of the recursive path ordering [12, Defn 4.22, 4.24] also follows. We then show how the theorem can be used in a number of specific examples.

Commonly, a rewrite system can be defined by taking a base system, known to be terminating, and adding new function symbols and rules to it. In §6 we show how our theorem can be used to prove termination in certain such cases, for example, where the new symbols and rules are those of the examples in §3.3 to §3.5 of [8]. A more complex variant of the result covers the example in [8, §3.6]. We then derive a path-ordering result in this context of incrementally defined systems.

The history of our work leading to these results is as follows. In [7] we described a proof of strong normalisation for a cut-elimination procedure in display logic. Since the procedure involved repeated transformation (or "rewriting") of a derivation tree, we realised that if we described a derivation tree as a term in a first-order langauge, we would get a proof of termination of a particular rewriting system. Determining the crucial properties of this rewriting system gave us Theorem 2 of [8]. Finally, noting the similarities and differences between this result and JGL1, Goubault-Larrecq's [19, Theorem 1], led us to our main result, Theorem 1.

Our proofs were formalised and machine-checked in the theorem prover Isabelle/HOL: see [10], directories `snabs,snlc`. This was particularly valuable for §6, where our initial paper proofs turned out to be wrong, as the choice of $\ll'_2$ was particularly difficult to get right. Further, the possible instantiation of a variable by a term headed by a symbol in either $\mathcal{F}_0$ or $\mathcal{F}_1$ complicated matters: see the discussion following [16, Proposition 1]. Also in §4.1 the Isabelle proof confirmed the validity of the rather tricky argument for the soundness of the mutually recursive definitions of SN, ρ and τ.

1.1. Notation, Terminology, Definitions and Basic Lemmas

We assume a set $\mathcal{U}$: in a TRS this would be the set of terms, but in the ARS setting we just call them "objects". For an irreflexive binary relation ρ, we will write $(r,t) \in \rho$, $(r,t) \in <_\rho$, $r <_\rho t$ or $t >_\rho r$ interchangeably. We prefer $>_\rho$ over the more traditional $\to_\rho$ because the latter is typically used in TRSs, our setting is more abstract than TRSs, and when we deal with TRSs we need to carefully distinguish between a relation and its closure under contexts. For a symbol that suggests a direction such as $<$, $\lhd$, $\prec$, $\precsim$ or $\ll$ we will write $(r,t) \in \lhd$, $(t,r) \in \rhd$, $r \lhd t$ or $t \rhd r$ interchangeably. We say r is *strongly normalising*, or is $\in SN$, (with respect to ρ) if there is no infinite descending sequence $r = r_0 >_\rho r_1 >_\rho r_2 >_\rho \ldots$ of objects, and ρ is *well-founded* (or *Noetherian*) if every $r \in SN$. We write $\leq_\rho$ or $\rho^=$, $<^+_\rho$ or ρ^+, and $<^*_\rho$ or ρ^* for the reflexive closure, the transitive closure and the reflexive transitive closure, respectively, of $<_\rho$. We write $\sigma \circ \rho$ for the relational composition of relations σ and ρ: that is, $(r,s) \in \sigma \circ \rho$ if there exists t such that $(r,t) \in \rho$ and $(t,s) \in \sigma$.

In our formal treatment in Isabelle/HOL we used the following inductive definition for the set SN of strongly normalising objects, and we proved, in the HOL logic, which is classical and contains the Axiom of Choice, that this definition is equivalent to the standard definition given above.

Definition 1 (Strongly Normalising – HOL) *For a reduction relation ρ, the set SN of strongly normalising objects is the (unique) smallest set of objects such that: if every object t to which s reduces is in SN then $s \in SN$.*

Our previous work [8], on term rewriting systems, dealt with the well-foundedness of the closure under context of a relation called ρ. In contrast, we are dealing here with an abstract reduction system, usually calling the reduction relation ρ. So concepts such as "strongly normalising", "reduction", etc, relate to ρ, and not, even when discussing structured terms, to the closure of ρ under context. For structured terms, ρ may be the closure under contexts of another relation σ, which may itself be the set of substitution instances of a set R of rewriting rules. Furthermore, in the ARS setting, we use an arbitrary relation where we used the immediate subterm relation in the TRS setting.

In [8] we defined the set *ISN* of "inductively strongly normalising" terms as the set of terms that are in *SN* if their immediate subterms are in *SN* [8, §2.2]. Clearly, $SN \subseteq ISN$. We now define `gindy` as a generalised notion of "inductively" for an arbitrary relation σ in place of the immediate subterm relation *isubt*. Use of `gindy` enables us to express the principle of well-founded induction succinctly: it says that if every object is in `gindy` σ S, and σ is well-founded, then every object is in S.

Definition 2 (`gindy`) *For a relation σ and set S, an object $t \in$ `gindy` σ S iff: if $\forall u.\ (u, t) \in \sigma \Rightarrow u \in S$, then $t \in S$.*

The notion of well-foundedness is generalised to that of a particular object being *accessible*, or in the *well-founded part*, of a binary relation: the constructive definition is that s is in the *well-founded part* of a relation $<$ if there is no infinite downward chain starting from s. This is generalised to the notion that *s bars S* in $<$ if every infinite downward chain, starting from s, contains a member of S. See [19] for a more detailed discussion of this. We now generalise `bars` to a function `gbars` where the members of a downward chain, until it meets S, must be in Q.

The inductive definition of `gbars` is:

Definition 3 (`gbars`) *For sets of objects Q and S, and relation σ, `gbars` σ Q S is the (unique) smallest set such that:*

(i) $S \subseteq$ `gbars` σ Q S
(ii) *if $t \in Q$ and $\forall u.\ (u, t) \in \sigma \Rightarrow u \in$ `gbars` σ Q S, then $t \in$ `gbars` σ Q S.*

The next lemma gives another characterisation of `gbars` which is provably equivalent in classical logic using the Axiom of Choice: $t \in$ `gbars` σ Q S iff every downward σ-chain starting from t is within Q until it hits S or it terminates.

Lemma 1 (`gbars`**-alternative**) *For sets of objects Q and S, and relation σ, object $t \in$ `gbars` σ Q S iff: for every downward σ-chain $t = t_0 >_\sigma t_1 >_\sigma t_2 >_\sigma \ldots$, either the chain is finite and all $t_i \in Q$, or for some member t_n of the chain, both $t_n \in S$ and $\{t_0, t_1, t_2, \ldots, t_{n-1}\} \subseteq Q$.*

Definition 4 records how `gbars` generalises the notions of "*S bars s* in σ" and of "s is *accessible* in σ, or s is in the *well-founded part* of σ" as defined in [19].

Definition 4 (***wfp***, `bars`)

(i) $s \in$ `bars` σ S *iff* $s \in$ `gbars` σ $\mathcal{U}$ S *("S* bars *s in σ")*
(ii) $s \in wfp\ \sigma$ *iff* $s \in$ `bars` σ $\emptyset$ *("s* accessible *in σ ").*

Thus $SN = wfp\ \rho$ and $ISN =$ `gindy` *isubt* SN. Now from Lemma 1 we get the following characterisation of `bars`, which was given as a definition in [19]: $s \in$ `bars` σ S if every infinite decreasing σ-sequence $s_0 >_\sigma s_1 >_\sigma s_2 >_\sigma \ldots$ meets S, ie, for some k, $s_k \in S$.

Lemma 2 relies on the `gbars`-induction principle, which is analogous to the principles of well-founded induction, and of `bars`-induction (see [19, Proposition 1]).

It is generated automatically by the Isabelle theorem prover from the inductive definition of `gbars` above. We write $\mathcal{P}\ s$ to mean that s satisfies property $\mathcal{P}$.

Proposition 1 (gbars-induction) *For sets Q and S, and any property $\mathcal{P}$, if*

(i) for every $s \in S$, we have $\mathcal{P}\ s$, and

(ii) for every $s \in Q$, if $\forall t.\ (t, s) \in \sigma \Rightarrow \mathcal{P}\ t$, then $\mathcal{P}\ s$

then every $s \in$ gbars $\sigma\ Q\ S$ satisfies $\mathcal{P}$.

Lemma 2

(i) $S =$ gbars σ (gindy $\sigma\ S$) S

(ii) $Q \subseteq$ gindy σ (gbars $\sigma\ Q\ S$)

Proof.

(i) $\subseteq$: this is trivial, from Definition 3(i), by letting Q be `gindy` $\sigma\ S$.
$\supseteq$: Let $\mathcal{P}\ s = s \in S$. We use Proposition 1 with $Q =$ `gindy` $\sigma\ S$. Condition (i) of Proposition 1 holds trivially, and condition (ii) is given by Definition 2.

(ii) Follows directly from Definitions 2 and 3(ii). □

Lemma 3

(i) if all objects are in gindy $\sigma\ S$, then bars $\sigma\ S = S$, whence, if σ is well-founded, then every object is in S, and

(ii) bars σ (wfp σ) = wfp σ

Proof.

(i) As $\mathcal{U} =$ `gindy` $\sigma\ S$, this follows from Lemma 2(i) and Definition 4(i). If σ is well-founded, then $\mathcal{U} = wfp\ \sigma =$ `bars` $\sigma\ \emptyset \subseteq$ `bars` $\sigma\ S$.

(ii) follows from (i) since, by Lemma 2(ii), every object is in `gindy` σ ($wfp\ \sigma$). □

2. The Termination Theorem

Given a reduction relation ρ, our general termination result requires relations $\lhd$ and $\ll$ which satisfy certain properties. These relations play a role similar to the relations $\lhd$ and $\ll$ in [19], and, where convenient, we express our conditions so as to enable easy comparisons with [19]. Most commonly, the relation $\lhd$ is instantiated to the immediate subterm relation, and $\ll$ is often some sort of approximation to the rewrite relation itself. The most general version of the properties that $\lhd$ and $\ll$ must satisfy is Condition 1(i) below, but in practice we often use the simpler and stronger conditions (ii) to (v). (Even weaker conditions than 1(i) are possible, since we could for example suppose that s also satisfies $\forall u \ll s.\ u \in$ `gindy` $\lhd$ *SN*: see the proof of Lemma 5 below).

Condition 1

(i) If $\forall s' \lhd s.\ s' \in SN$, then $s \in$ bars ρ (gbars $\lhd$ $\{u \mid u \ll s\}$ SN)

(ii) For all $(t, s) \in \rho$, if $\forall s' \lhd s.\ s' \in SN$, then $t \in$ gbars $\lhd$ $\{u \mid u \ll s\}$ SN

(iii) For all $(t, s) \in \rho$, $t \in$ gbars $\lhd$ $\{u \mid u \ll s\}$ $\{v \mid (v, s) \in (\lhd \circ \rho)\}$

(iv) $\lhd$ is well-founded and, for all $(t, s) \in \rho$, if $\forall s' \lhd s.\ s' \in SN$, then, for all $t' \lhd^* t$, either $t' \in SN$ or $t' \ll s$

(v) $\lhd$ *is well-founded and, for all* $(t, s) \in \rho$ *and all* $t' \lhd^* t$, *either* $(t', s) \in (\lhd \circ \rho^*)$ *or* $t' \ll s$.

Lemma 4 *Each of Conditions 1(ii) to (v) implies Condition 1(i) for all* s.

Proof. It is easy to see that Condition 1(ii) implies Condition 1(i) for all s.

To show that Condition 1(iv) implies Condition 1(ii), assume (iv) holds. Then, as $\lhd$ is well-founded, there is no infinite descending $\lhd$-chain. Any descending $\lhd$-chain from t is contained in $\{t' \mid t' \ll s\} \cup SN$. A fortiori, members of such a chain are contained in $\{t' \mid t' \ll s\}$ until the chain reaches a member of SN. That is, $t \in \texttt{gbars} \lhd \{u \mid u \ll s\}\ SN$, and so (ii) holds.

To show that Condition 1(v) implies Condition 1(iv), and likewise, that Condition 1(iii) implies Condition 1(ii), assume that $\forall s' \lhd s.\ s' \in SN$. Then, if $(t', s) \in (\lhd \circ \rho^*)$ because $t' <^*_\rho s' \lhd s$, then we have $s' \in SN$ and so $t' \in SN$. Note that in Condition 1(iii) we could have $(\lhd \circ \rho^*)$ in place of $(\lhd \circ \rho)$. □

Our key lemma is Lemma 5. We thank an unnamed referee for pointing out that our proof resembles the proof by Buchholz [6] of the well-foundedness of the lexicographic path ordering, although it was obtained independently.

Lemma 5 *If object* s *satisfies Condition 1(i), then* $s \in \texttt{gindy} \ll (\texttt{gindy} \lhd SN)$.

Proof. Given s, assume that ρ, $\lhd$ and $\ll$ satisfy Condition 1(i) and that

(a) $\forall u \ll s.\ u \in \texttt{gindy} \lhd SN$.

We then need to show $s \in \texttt{gindy} \lhd SN$, so we assume

(b) $\forall s' \lhd s.\ s' \in SN$

and we show that $s \in SN$. By Lemma 3(ii), it suffices to show $s \in \texttt{bars}\ \rho\ SN$.

The antecedent of Condition 1(i) holds by assumption (b), and so $s \in \texttt{bars}\ \rho\ (\texttt{gbars} \lhd \{u \mid u \ll s\}\ SN)$. As bars is monotonic in its second argument, it is enough to show $\texttt{gbars} \lhd \{u \mid u \ll s\}\ SN \subseteq SN$. As $\{u \mid u \ll s\} \subseteq \texttt{gindy} \lhd SN$ by assumption (a), and as gbars is monotonic in its second argument, $\texttt{gbars} \lhd \{u \mid u \ll s\}\ SN \subseteq \texttt{gbars} \lhd (\texttt{gindy} \lhd SN)\ SN$. By Lemma 2(i), $\texttt{gbars} \lhd (\texttt{gindy} \lhd SN)\ SN = SN$.

So we have $s \in SN$. Thus, discharging assumptions (b) and then (a), we have $s \in \texttt{gindy} \lhd SN$, and then $s \in \texttt{gindy} \ll (\texttt{gindy} \lhd SN)$. □

We now identify the conditions that guarantee that every object is in SN.

Theorem 1 *Suppose that* ρ, $\lhd$ *and* $\ll$ *satisfy Condition 1(i) for all* s *and*

(i) every object is in $\texttt{bars} \ll (\texttt{gindy} \lhd SN)$, *and*

(ii) every object is in $\texttt{bars} \lhd SN$.

Then ρ *is well-founded.*

Proof. As ρ, $\lhd$ and $\ll$ satisfy Condition 1(i), every $s \in \texttt{gindy} \ll (\texttt{gindy} \lhd SN)$ by Lemma 5. Then, for any u, if $u \in \texttt{bars} \ll (\texttt{gindy} \lhd SN)$ then Lemma 3(i) gives $u \in \texttt{gindy} \lhd SN$. Thus every $u \in \texttt{gindy} \lhd SN$. Then, for any v, if $v \in \texttt{bars} \lhd SN$ then Lemma 3(i) gives $v \in SN$. Thus every $v \in SN$: that is, ρ is well-founded. □

If Condition 1(ii) holds, then it also holds if we augment ρ to contain $\lhd$. Thus, for fixed $\lhd$, Theorem 1 does *not* provide a universal method of proving termination because it is possible that ρ is well-founded but $\rho \cup \lhd$ is not.

However, if ρ is well-founded and we can choose $\lhd$ so that $\lhd \subseteq \rho$, then Theorem 1 can be applied trivially. Let $\ll \; = \; \lhd \; = \; \rho^+$, which is well-founded. Then even Condition 1(v) applies. Clearly also, conditions (i) and (ii) of Theorem 1 apply as $\ll$ and $\lhd$ are well-founded. That is, Theorem 1 is, trivially, a universal result for proving termination (as are several other orderings in the literature).

Most commonly the conditions of Theorem 1 will be satisfied by choosing $\ll$ and $\lhd$ to be well-founded relations. At this point we mention a lemma which we will use several times, for proving that a relation is well-founded.

Lemma 6 *Assume that r and s are well-founded relations. Then each of the following conditions is sufficient for $\phi \cup \psi$ to be well-founded:*

(a) $\phi \circ \psi \;\subseteq\; \psi^* \circ \phi$
(b) $\phi \circ \psi \;\subseteq\; \psi \circ \phi^*$
(c) $\phi \circ \psi \;\subseteq\; \phi \cup \psi$
(d) $\phi \circ \psi \;\subseteq\; (\psi \circ (\phi \cup \psi)^*) \cup \phi$.

Of these, the last is from Doornbos & von Karger [15], and is implied by each of the others, which are in earlier results discussed and cited in [15].

3. Applications to Abstract Rewrite Systems

3.1. Generalising Goubault-Larrecq's General Theorem for ARSs

We show that JGL1, ie, Goubault-Larrecq's [19, Theorem 1], which itself generalizes many results in the literature, is a special case of our Theorem 1. Note that [19] uses $<$ where we use ρ. We first require two lemmas.

Lemma 7 *Given set S and object s, suppose for all t that, if $(t, s) \in \rho$, then*

(i) $t \in S$, *or*
(ii) $s \gg t$ *and, for every* $u \lhd t$, *either* $(u, s) \in \rho$ *or* $u \in S$.

Assume $\lhd$ is well-founded. Then $(t, s) \in \rho$ implies $t \in \texttt{gbars} \lhd \{x \mid x \ll s\}\ S$.

Proof. Let $(t, s) \in \rho$. We prove this result for t by well-founded induction on $\lhd$, so assume that, for all $v \lhd t$, if $(v, s) \in \rho$ then $v \in \texttt{gbars} \lhd \{x \mid x \ll s\}\ S$.

We consider the two cases (i) and (ii) as above. Firstly, if $t \in S$, then $t \in \texttt{gbars} \lhd \{x \mid x \ll s\}\ S$ by Definition 3(i). Secondly, if (ii) holds, we show that $t \in \texttt{gbars} \lhd \{x \mid x \ll s\}\ S$ using Definition 3(ii). We have $t \ll s$, and for any $u \lhd t$, there are again two cases. In the first case, $(u, s) \in \rho$ and so, by the inductive hypothesis, $u \in \texttt{gbars} \lhd \{x \mid x \ll s\}\ S$. In the second case, $u \in S$ and so $u \in \texttt{gbars} \lhd \{x \mid x \ll s\}\ S$ by Definition 3(i). □

Lemma 8 *Suppose that, whenever $(t, s) \in \rho$, either*

(i) for some object u, $s \rhd u$ and $u \geq_\rho t$, or
(ii) $s \gg t$ and, for every $u \lhd t$, $s >_\rho u$.

If $\lhd$ is well-founded, then Condition 1(ii) holds.

Proof. We use Lemma 7 with $S = \{v \mid \exists x.\ s \rhd x \text{ and } x \geq_\rho v\}$. To show Condition 1(ii), let $(t, s) \in \rho$, and suppose that $\forall s' \lhd s.\ s' \in SN$. By Lemma 7, $t \in \texttt{gbars} \lhd \{x \mid x \ll s\}\ S$. We show $S \subseteq SN$. Let $s \rhd x$ and $x \geq_\rho v$. Then

$x \in SN = wfp\ \rho$ and so $v \in SN$. Thus, by the obvious monotonicity of `gbars`, $t \in \texttt{gbars} \lhd \{x \mid x \ll s\}\ SN$, as required for Condition 1(ii). □

Corollary 1 *JGL1, that is, [19, Theorem 1], holds.*

Proof. JGL1 says: if Property 1 and conditions (*iii*) and (*iv*) (as given in [19]) hold, then ρ is well-founded.

Condition (*iv*) of [19] is just condition (i) of our Theorem 1 because $\underline{SN}$ of [19] is $\texttt{gindy} \lhd SN$, and if some $u \lhd s \notin SN$, then $s \in \underline{SN}$. Thus the requirement that "if every $u \lhd s$ is in SN" in the statement of JGL1 is redundant, although its counterpart is needed in the statement of [19, Theorem 2].

Condition (*iii*) of [19] says that $\lhd$ is well-founded. Then, for any object v and set S of objects, $v \in \texttt{bars} \lhd S$, and so condition (ii) of our Theorem 1 follows.

Property 1 of [19] says that for $(t, s) \in \rho$, either (i) or (ii) of Lemma 8 holds. Finally, Lemma 8 shows that if $\lhd$ is well-founded, as ensured by condition (*iii*) of [19], then Property 1 implies Condition 1(ii), whence Condition 1(i) holds.

Thus all the conditions of our Theorem 1 hold, so ρ is well-founded. □

We explore the extent to which, conversely, JGL1 implies our Theorem 1. We discuss in detail only whether our Conditions 1(i) to (v) imply Property 1 of [19]. First we note a sort of converse to Lemma 7: if $(t, s) \in \rho \Leftrightarrow t \in \texttt{gbars} \lhd \{x \mid x \ll s\}\ S$ then, by the definition of `gbars`, (i) or (ii) of Lemma 7 hold, even after deleting "*or* $u \in S$" from (ii).

Suppose Condition 1(iii) holds: that is, with $S = \{v \mid (v, s) \in (\lhd \circ \rho)\}$, we have $(t, s) \in \rho \Rightarrow t \in \texttt{gbars} \lhd \{x \mid x \ll s\}\ S$. Since `gbars` is monotonic in its third argument and S is monotonic in ρ, we can enlarge ρ so that Condition 1(iii) holds as an equivalence, giving (i) and (ii) of Lemma 8 (*i.e.* Property 1 of [19]).

That is, JGL1 can be used to prove a weaker version of our Theorem 1 in which Condition 1(iii), rather than Condition 1(i), is assumed, and it is assumed that $\lhd$ is well-founded. On the other hand, in Sections 4.1 and 4.2, the proofs of termination use our Theorem 1, and, in particular, use Condition 1(ii). The difference between Condition 1(ii) and Condition 1(iii) is crucial to these proofs, which shows that JGL1 is a special case of our Theorem 1.

3.2. Application to the General Path Ordering

The definition of the general path ordering of Dershowitz & Hoot [14] presupposes terms in a first-order language, but the ordering is not necessarily monotonic. Therefore we deal with it in the context of an ARS. As noted by Goubault-Larrecq [19], JGL1 can be used to show that the general path ordering is well-founded. We now show how that ordering fits our conditions for termination of an ARS presented earlier. The following treatment is similar to that of the recursive path orderings in our [8, §3.7], but that applied only to the cases where the derived ordering is monotonic (ie, closed under context).

The general path ordering $<_{gpo}$ is defined as below where $s = f(s_1, \ldots, s_m)$ and $t = g(t_1, \ldots, t_n)$, and $\Lambda(<_{gpo})$ (or $<_\Lambda$) is an ordering derived from $<_{gpo}$, where Λ

satisfies conditions given later. We omit defining $<_{gpo}$-equivalent terms.

$$\frac{s_i \geq_{gpo} t}{s >_{gpo} t}(G1) \qquad \frac{s >_\Lambda t \qquad \forall i \in \{1, \ldots, n\}.\ s >_{gpo} t_i}{s >_{gpo} t}(G2)$$

In practice, $<_\Lambda$, which is derived from $<_{gpo}$, depends on $<_{gpo}$ applied only to smaller terms, ie, determining whether $s >_\Lambda t$ depends on whether $s' >_{gpo} x$ only for proper subterms s' of s (and arbitrary terms x). Thus the above rules can be seen as mutually recursive definitions of $<_{gpo}$ and $<_\Lambda$.

However in the formal Isabelle proof a different approach was taken to ensure a consistent definition: $<_{gpo}$ is an inductively defined set, which means that $s >_{gpo} t$ if and only if it can be shown so using the rules above. This approach requires that $<_\Lambda$ be a monotonic function of $<_{gpo}$.

As in [8], we define an auxiliary function *fwf* for "from well-founded" which maps a binary relation σ to a binary relation *fwf* σ as below:

Definition 5 *Given a relation σ, $(t, s) \in$ fwf σ iff $(t, s) \in \sigma$ and $s \in$ wfp σ.*

Clearly *fwf* ρ is well-founded, regardless of whether ρ is.

We can let $\lhd$ be either the immediate subterm relation or the proper subterm relation — in the latter case, the required condition (c) below is weaker. We now list the conditions on Λ required for the Isabelle proof:

(a) Λ is a monotonic function
(b) if σ is well-founded then $\Lambda(\sigma)$ is well-founded
(c) if all $s' \lhd s$ are in *wfp* σ and if $(t, s) \in \Lambda(\sigma)$, then $(t, s) \in \Lambda(\textit{fwf}\ \sigma)$.

In practice, condition (c) means that $\Lambda(\sigma)$ depends on σ only as follows: whether $(t, s) \in \Lambda(\sigma)$ depends solely upon whether $(x, s') \in \sigma$ for $s' \lhd s$, and arbitrary x, although $\Lambda(\sigma)$ can also depend on, for example, a well-founded ordering on the function symbols $\{f, g, \ldots\}$.

Lemma 9 *If $s >_{gpo} t$ and t' is a subterm of t then $s >_{gpo} t'$.*

Proof. It is enough to show that if $s >_{gpo} t = g(t_1, \ldots, t_n)$ then $s >_{gpo} t_j$ for any $j \in \{1, \ldots, n\}$. We show it by induction on the size (or structure) of s. If $s >_{gpo} t$ via rule $(G2)$, then it is immediate that $s >_{gpo} t_j$. If $s = f(s_1, \ldots, s_m) >_{gpo} t$ via rule $(G1)$, then for some s_i, either $s_i >_{gpo} t$ and so $s_i >_{gpo} t_j$ by induction, or $s_i = t >_{gpo} t_j$ by rule $(G1)$, and then $s >_{gpo} t_j$ by rule $(G1)$. □

Theorem 2 *The general path ordering $<_{gpo}$ is well-founded.*

Proof. Define $\ll\ = \Lambda(\textit{fwf}\,(<_{gpo}))$. Then, as $\textit{fwf}\,(<_{gpo})$ is well-founded, $\ll$ is well-founded by condition (b) above. As $\lhd$ is well-founded, conditions (i) and (ii) of our Theorem 1 are satisfied.

Then, to show Condition 1(iv), suppose for all $s' \lhd s$ that $s' \in SN = \textit{wfp}\,(<_{gpo})$, $t <_{gpo} s$, and t' is a subterm of t. Then Lemma 9 gives $t' <_{gpo} s$. If $t' <_{gpo} s$ by rule $(G1)$, then $s_i \lhd s$ and $t' <^*_{gpo} s_i$, so s_i and t' are in *SN*. If $t' <_{gpo} s$ by rule $(G2)$, then $(t', s) \in \Lambda(<_{gpo})$. By condition (c), $(t', s) \in \Lambda(\textit{fwf}\,(<_{gpo})) = \ll$. Thus Condition 1(iv) is satisfied, and so $<_{gpo}$ is well-founded. □

An appropriate choice of Λ can give us either the lexicographic or multiset path orderings for structured terms [12, Defn 4.22, 4.24]. To do so we let $f(s_1, \ldots, s_m) >_\Lambda$

$g(t_1, \ldots, t_n)$ iff either $f > g$ (according to a given well-founded ordering on function symbols) or $(s_1, \ldots, s_m) > g(t_1, \ldots, t_n)$ in the derived lexicographic or multiset ordering derived from $>_{gpo}$. Such a Λ satisfies properties (a) to (c) above.

These examples of Λ also have the property that:

$$\text{if } (s_i', s_i) \in \sigma \text{ then } (f(s_1, \ldots, s_i', \ldots, s_n), f(s_1, \ldots, s_i, \ldots, s_n)) \in \Lambda(\sigma)$$

This property ensures that $\Lambda(\sigma)$ is closed under context, but is not needed for this more general treatment.

4. Application to Typed Combinators

In [19] Goubault-Larrecq concludes that "Theorem 1 seems to be insufficient to show that every simply-typed λ-term terminates". He therefore takes notions like "reducibility" and "the substitution of terms for variables" from the classical strong normalisation proof of the simply-typed λ-calculus [18] and generalises them to obtain his Theorem 2 for termination of higher-order path orderings.

As the λ-calculus can be imitated by using the combinators S, K, I a related problem is to prove the strong normalisation for well-typed combinator terms. This result follows easily from the strong normalisation of β-reduction. But to prove the converse, that strong normalisation of β-reduction follows from that of well-typed combinator terms, is not so easy: one needs a translation from λ-terms to combinator terms that preserves reducibility, such as of Akama [1].

We now describe two ways to use our Theorem 1 to prove strong normalisation of well-typed combinator terms. These proofs resemble classic "reducibility" arguments, but do not handle substitution of terms for variables. By Akama [1, Theorem 2.2], this is enough to show strong normalisation of β-reduction.

Thus, the full power of the much more complex [19, Theorem 2] is not necessary for these tasks. However, we have been unable to prove termination of typed combinators using our [8, Theorem 2] as suggested by an anonymous referee. Also, although we could prove strong normalisation of the simply-typed λ-calculus by adapting the proof of our Theorem 1, we could not prove it as a corollary of our Theorem 1.

4.1. Reduction of Typed Combinator Expressions

Of the usual combinators, the problematic ones are $S\,f\,g\,x = f\,x\,(g\,x)$ and $W\,f\,x = f\,x\,x$, since their right-hand-sides duplicate x. Thus, in the untyped setting, these do not satisfy strong normalisation: for example, $(SII)(SII) \longrightarrow^+ (SII)(SII)$, $(WI)(WI) \longrightarrow^+ (WI)(WI)$, and $WWW \longrightarrow WWW$.

In the typed setting, we can use our Theorem 1 to prove strong normalisation for combinators like S, I, K, B, C, W, but we must actually add extra rules to do so. Note that some rules are only applied to the whole term, whereas others can also be applied to any subterm. We use $\alpha, \beta, \gamma, \ldots$ for types, and the notation $<$ on types for the transitive relation given by: $\alpha < (\alpha \to \beta)$ and $\beta < (\alpha \to \beta)$. Let $\mathit{tyof}\, t$ denote the type of t, and let $t <_{ty} s$ mean $\mathit{tyof}\, t < \mathit{tyof}\, s$.

We show in detail how to handle only the S combinator, but many other combinators like I, K, B, C, W can be dealt with similarly, and the proof holds for the single system containing all these combinators. We define the relations σ and τ (as inductively defined sets) by rules, as shown below for the combinator S. Then let $\rho = \mathit{ctxt}\ \sigma \ \cup\ \tau$. Note that the extra rules (2), (3) and (4) are added only because the proof uses them: for clearly, if $\rho \supseteq \rho'$, and ρ terminates, then so does ρ'. These rules are motivated by the relation $\lhd^\circ$ of [19, §5]:

$$
\begin{aligned}
Sfgx &>_\sigma fx(gx) & & (1)\\
Sfg &>_\tau fx(gx) & \text{if } x \in \mathit{SN} & \quad (2)\\
Sf &>_\tau fx(gx) & \text{if } g, x \in \mathit{SN} & \quad (3)\\
S &>_\tau fx(gx) & \text{if } f, g, x \in \mathit{SN} & \quad (4)
\end{aligned}
$$

with types $S : (\alpha \to \beta \to \gamma) \to (\alpha \to \beta) \to \alpha \to \gamma$, $f : \alpha \to \beta \to \gamma$, $g : \alpha \to \beta$, $x : \alpha$.

Note that "SN" means with respect to ρ: that is, τ and ρ (but *not* σ) are being defined, indirectly, in terms of themselves. However the rule (1) preserves the type of a term: so, when it is applied to a subterm, the whole term remains well-typed. The rules for τ change a well-typed term into a well-typed term of $<$-smaller type. Further, where a rule for reducing a term s depends on another term s' being in SN, then $s' <_{ty} s$. For example, in rule (3), we have $Sf : (\alpha \to \beta) \to \alpha \to \gamma$, where g and x have $<$-smaller types, $g : \alpha \to \beta$ and $x : \gamma$.

That is, for $s : \alpha$, the set $\{t \mid (t, s) \in \rho\}$ depends on σ and on $\{t \mid (t, s) \in \tau\}$, which depends on $\{s' \in \mathit{SN} \mid \mathit{tyof}\ s' < \alpha\}$. For given $\beta < \alpha$, the set $\{s' \in \mathit{SN} \mid s' : \beta\}$ depends on $\{(x, s'') \in \rho \mid \mathit{tyof}\ s' \leq \beta\}$. This ensures a consistent definition, as for $\lhd^\circ$ in [19]. In effect, whether $(t, s) \in \rho$, $(t, s) \in \tau$ and $u \in \mathit{SN}$ are defined inductively on the types of s and of u.

We say $t <_{sn1} s$ if $(t, s) \in \mathit{ctxt}\ \sigma$ via a reduction in an immediate subterm which is itself in $\mathit{wfp}\,(\mathit{ctxt}\ \sigma)$: that is, where t and s differ only in corresponding immediate subterms t' and s' with $(t', s') \in \mathit{ctxt}\,\sigma$ and $s' \in \mathit{wfp}\ (\mathit{ctxt}\,\sigma)$. (Note that the immediate subterms of $f\ x\ y$ are $f\ x$ and y).

Lemma 10 *Let $\ll\ = <_{ty} \cup <_{sn1}$, and let $\lhd$ be the immediate subterm relation. Then Condition 1(ii) holds.*

Proof. Let $(t, s) \in \rho$ and assume $\forall s' \lhd s.\ s' \in \mathit{SN}$. If $(t, s) \in \rho$ via rule (2) (where $s = Sfg$), we have $g \in \mathit{SN}$, so $(t, Sf) \in \rho$ by rule (3). As $Sf \lhd s$, we have $Sf \in \mathit{SN}$, so $t \in \mathit{SN} \subseteq \texttt{gbars} \lhd \{u \mid u \ll s\}\ \mathit{SN}$. Similar arguments hold where $(t, s) \in \sigma \subseteq \rho$ by rule (1), and where $(t, s) \in \tau \subseteq \rho$ via rule (3).

If $(t, s) \in \tau \subseteq \rho$ via rule (4), then we see that the subterms f, g and x of t are in SN, while the subterms $fx : \beta \to \gamma$, $gx : \beta$ and $t = fx(gx) : \gamma$ are of $<$-smaller type than S. Thus any $\lhd$-descending chain from t consists of terms in $\{u \mid u \ll s\}$ until reaching a term in SN. That is, $t \in \texttt{gbars} \lhd \{u \mid u \ll s\}\ \mathit{SN}$.

Finally, in the case where $(t, s) \in \mathit{pctxt}\ \sigma$, we have $t' \lhd t$ and $s' \lhd s$ such that $(t', s') \in \mathit{ctxt}\,\sigma$. Since $s' \in \mathit{SN} \subseteq \mathit{wfp}\,(\mathit{ctxt}\ \sigma)$, we have $t <_{sn1} s$ and $t \ll s$. Now consider any $t'' \lhd t$. Either $t'' = t'$ and $(t', s') \in \mathit{ctxt}\,\sigma$ as just discussed, in which case $s' \in \mathit{SN}$ and so $t' \in \mathit{SN}$, or t'' is an immediate subterm of s not affected by the reduction from s to t, whence $t'' \in \mathit{SN}$.

Therefore $t \in \texttt{gbars} \lhd \{u \mid u \ll s\}\ \mathit{SN}$. □

Theorem 3 *Every term is strongly normalising.*

Proof. We use our Theorem 1, and Lemmas 4 and 5. Apart from Lemma 10, we need conditions (i) and (ii) of our Theorem 1. Condition (ii) holds as $\lhd$ is well-founded. Finally, to show condition (i), we show that $\ll$ is well-founded. Clearly the "smaller type" relation $<$ is well-founded, and it is easy to show (analogously to Theorem 6) that $<_{sn1}$ is well-founded. Then, clearly, $<_{ty} \circ <_{sn1} \subseteq <_{ty}$, and so, by Lemma 6(a), $<_{ty} \cup <_{sn1}$ is well-founded. □

Our rules (2), (3) and (4) were suggested by the definition of $\lhd^\circ$ given just below Remark 13 in [19]. As in [19], therefore, there is a resemblance between our proof and the classic reducibility argument: we have, for example, that for $S\ f\ g$ to be in *SN*, it is necessary that for all $x \in \mathit{SN}$, $S\ f\ g\ x \in \mathit{SN}$, which resembles the condition for reducibility in [18, §6.1]. Likewise, reducibility and our *SN* are both defined by induction on the type.

4.2. A Second Proof for Typed Combinator Expressions

We now present another way of using our Theorem 1 to prove the same result. This proof was suggested by a presentation of the classic reducibility argument given us by an unnamed referee. It is of independent interest since, unlike the proof in §4.1, it uses a relation $\lhd$ which is distinct from the usual immediate subterm relation. We define $\lhd$ and the reduction relation ρ. Again, it is understood that terms are well-typed. Combinators other than S could be included also.

$$N_i \lhd M N_1 \dots N_i \dots N_n \quad \text{for } 1 \le i \le n \tag{5}$$

$$M >_\rho MN \quad \text{if } N \in \mathit{SN} \tag{6}$$

$$Sfgxy_1 \dots y_n >_\sigma fx(gx)y_1 \dots y_n \tag{7}$$

$$\sigma \subseteq \rho \tag{8}$$

$$(x'_i, x_i) \in \mathit{ctxt}\ \sigma \Rightarrow \quad f x_1 \dots x_i \dots x_n >_\rho f x_1 \dots x'_i \dots x_n \tag{9}$$

Note that rules (7) to (9) together give $\mathit{ctxt}\ \sigma \subseteq \rho$. These definitions are sound as before, since again, a reduction preserves type or gives a result of $<$-smaller type, and reduction from s is defined involving *SN* terms of $<$-smaller type. Note that, by rule (6), if $M, N \in \mathit{SN}$ then $MN \in \mathit{SN}$.

For this proof we define $f x_1 \dots x_i \dots x_n >_{sn1} f x_1 \dots x'_i \dots x_n$ where $(x'_i, x_i) \in \mathit{ctxt}\ \sigma$ and $x_i \in \mathit{wfp}\,(\mathit{ctxt}\ \sigma)$. That is, as before, $t <_{sn1} s$ if $(t, s) \in \mathit{ctxt}\ \sigma$ by means of reduction in a "$\lhd$-subterm" which is itself in $\mathit{wfp}\,(\mathit{ctxt}\ \sigma)$.

Also as before, let $\ll\ = <_{ty} \cup <_{sn1}$.

Theorem 4 *Every term is strongly normalising.*

Proof. We first show that Condition 1(ii) holds. Let $(t, s) \in \rho$ and assume that $\forall u \ll s.\ u \in \texttt{gindy} \lhd \mathit{SN}$, and $\forall v \lhd s.\ v \in \mathit{SN}$. If $(t, s) \in \rho$ via rule (7), we have f, g, x and each $y_i \in \mathit{SN}$, so the combination $fx(gx)y_1 \dots y_n \in \mathit{SN}$.

If $(t, s) = (MN, M) \in \rho$ via rule (6), then $t <_{ty} s$, so $t \ll s$, and, for $K \lhd MN$, either $K = N$ which is in *SN*, or $K \lhd M$ and so $K \in \mathit{SN}$.

Finally, where $(t, s) \in \rho$ via rule (9), the argument is similar to that before: $t <_{sn1} s$, and for $y \lhd t$, there is $x \lhd s$ such that $y \leq_\rho x$, so $y \in \mathit{SN}$.

That is, in each case, $t \in \texttt{gbars} \lhd \{u \mid u \ll s\}\ \mathit{SN}$, so Condition 1(ii) holds.

From Condition 1(ii), we prove that every term is in SN as in Theorem 3. □

4.3. Strong Normalization for System T

We can also develop the previous proofs to handle the strong normalization of System T, though in each case there is a complication. System T [2] consists of the system of typed combinators as above, with a distinguished type *nat*, the usual two constructors for this type, $\mathit{Zero} : \mathit{nat}$ and $\mathit{Succ} : \mathit{nat} \to \mathit{nat}$, and the general recursive function R defined by

$$R\ \mathit{Zero}\ f\ z = z \qquad\qquad R\ (\mathit{Succ}\ n)\ f\ z = f\ n\ (R\ n\ f\ z)$$

So to extend the first proof to this situation we add the rules

$$R\ \mathit{Zero}\ f\ z >_\sigma z \tag{10}$$
$$R\ (\mathit{Succ}\ n)\ f\ z >_\sigma f\ n\ (R\ n\ f\ z) \tag{11}$$
$$R\ (\mathit{Succ}\ n)\ f >_\tau f\ n\ (R\ n\ f\ z) \qquad \text{if } z \in \mathit{SN} \tag{12}$$
$$R\ (\mathit{Succ}\ n) >_\tau f\ n\ (R\ n\ f\ z) \qquad \text{if } f, z \in \mathit{SN} \tag{13}$$
$$\mathit{Succ}\ n >_\tau n \tag{14}$$

We now prove Lemma 10 for this extended system, using the same $\lhd$ and $\ll$.

Lemma 11 *Consider the system of §4.1, extended by rules (10) to (14). Let* $\ll$ *be* $<_{ty} \cup <_{sn1}$*, and* $\lhd$ *the immediate subterm relation. Then Condition 1(ii) holds.*

Proof. Let $(t, s) \in \rho$ and assume $\forall s' \lhd s.\ s' \in \mathit{SN}$. If $(t, s) \in \rho$ via rule (10) then $t = z \lhd s$, so $t \in \mathit{SN}$. If $(t, s) \in \rho$ via rule (11) or rule (12) the argument is as for rules (1) to (3).

If $(t, s) \in \rho$ via rule (13) then first note that as $\mathit{Succ}\ n \in \mathit{SN}$, and $\mathit{Succ}\ n >_\tau n$ by rule (14), so $n \in \mathit{SN}$. We now need to consider any $\lhd$-descending chain from $t = f\ n\ (R\ n\ f\ z)$, and consider the subterms t' on such a chain. Since $s = R\ (\mathit{Succ}\ n)$ is of type $(\mathit{nat} \to \alpha \to \alpha) \to \alpha \to \alpha)$, we see that $t : \alpha$ and its subterms $f\ n : \alpha \to \alpha$, $R\ n\ f\ z : \alpha$ and $R\ n\ f : \alpha \to \alpha$ are of $<$-smaller type than S. Also $R\ n <_{sn1} R\ (\mathit{Succ}\ n)$, since $R\ n$ is obtained by reducing the strongly normalizing subterm $(\mathit{Succ}\ n)$. So for all these subterms t', we have $t' \ll s$. Finally, we have that f, z, n and R are in SN. Thus any $\lhd$-descending chain from t consists of terms in $\{u \mid u \ll s\}$ until meeting a term in SN; that is, $t \in \texttt{gbars} \lhd \{u \mid u \ll s\}\ \mathit{SN}$. □

Finally, every term is strongly normalising, proved exactly as in Theorem 3.

4.4. Adapting the second proof to System T

We now show how to adapt the proof of §4.2 to System T. We add the following rules to those given earlier, ie, to rules (6) to (9):

$$R\ \mathit{Zero}\ f\ z\ y_1 \ldots y_k >_\sigma z\ y_1 \ldots y_k \tag{15}$$
$$R\ (\mathit{Succ}\ n)\ f\ z\ y_1 \ldots y_k >_\sigma f\ n\ (R\ n\ f\ z)\ y_1 \ldots y_k \tag{16}$$
$$\mathit{Succ}\ n >_\tau n \tag{17}$$

We define $\lhd$ as before, ie, by rule (5), but we need a different definition of $\ll$, as follows where $<_{sn1}$ is defined as in §4.2, $f\ \overline{x'}$ means $fx_1 \ldots x'_i \ldots x_m$, $f\ \overline{x}$ means $fx_1 \ldots x_i \ldots x_m$, $f\ \overline{x}\ \overline{y}$ means $f\ \overline{x}\ y_1 \ldots y_k$, and applications are well-typed.

$$MN \ll M \tag{18}$$

$$f\ \overline{x'} \ll f\ \overline{x}\ \overline{y} \qquad \text{where } f\ \overline{x'} <_{sn1} f\ \overline{x} \tag{19}$$

Lemma 12 $\ll$ *is well-founded.*

Proof. We use Lemma 6(d), with ϕ and ψ being the relations given by (18) and (19) respectively. Then we have the following cases for $u >_\phi v >_\psi w$:

- $f\ \overline{x} >_\phi f\ \overline{x}\ y >_\psi f\ \overline{x}\ y'$, when $u >_\phi w$
- $f\ \overline{x} >_\phi f\ \overline{x}\ y >_\psi f\ \overline{x'}\ y$, when $u >_\psi v' >_\phi w$
- $f\ \overline{x}\ \overline{w} >_\phi f\ \overline{x}\ \overline{w}\ y >_\psi f\ \overline{x'}$ when $u >_\psi w$

so in each case $(w, u) \in (\psi \circ (\phi \cup \psi)^*) \cup \phi$.

Since $<_{sn1}$ is well-founded it follows easily that ψ is well-founded, and ϕ is well-founded since $M\ N <_{ty} M$. Thus $\phi \cup \psi = \ll$ is well-founded. □

Theorem 5 *Every term is strongly normalising.*

Proof. In this case we cannot show that Condition 1(ii) holds, but we can show that every term is in $\texttt{gindy} \ll (\texttt{gindy} \lhd \mathit{SN})$. Let $(t, s) \in \rho$ and assume that $\forall u \ll s.\ u \in \texttt{gindy} \lhd \mathit{SN}$, and $\forall v \lhd s.\ v \in \mathit{SN}$. If $(t, s) \in \rho$ via rule (7), then, as in the proof of Theorem 4 we have f, g, x and each $y_i \in \mathit{SN}$, so the combination $fx(gx)y_1 \ldots y_k \in \mathit{SN}$.

If $(t, s) = (MN, M) \in \rho$ via rule (6), then $t \ll s$, so $t \in \texttt{gindy} \lhd \mathit{SN}$. Then for $K \lhd MN$, either $K = N$ which is in SN, or $K \lhd M$ and so $K \in \mathit{SN}$. Thus $t \in \mathit{SN}$.

If $(t, s) \in \rho$ via rule (15), then z and each $y_i \in \mathit{SN}$, so $t = z\ y_1 \ldots y_k \in \mathit{SN}$.

If $(t, s) \in \rho$ via rule (16), then we have $f, z, \mathit{Succ}\ n$ and each $y_i \in \mathit{SN}$, and so by rule (17), $n \in \mathit{SN}$. Thus, to get $t = f\ n\ (R\ n\ f\ z)\ y_1 \ldots y_k$, we need only show $R\ n\ f\ z \in \mathit{SN}$. Now $R\ n\ f\ z \ll R\ (\mathit{Succ}\ n)\ f\ z\ y_1 \ldots y_k = s$, so $R\ n\ f\ z \in \texttt{gindy} \lhd \mathit{SN}$. Now, for $t' \lhd R\ n\ f\ z$, $t' = n, f$ or z and so $t' \in \mathit{SN}$. Therefore $R\ n\ f\ z \in \mathit{SN}$.

If $(t, s) \in \rho$ via rule (17), then we have $t = n \in \mathit{SN}$.

Finally, where $(t, s) \in \rho$ via rule (9), $t <_{sn1} s$, so $t \in \texttt{gindy} \lhd \mathit{SN}$. Then for $y \lhd t$, there is $x \lhd s$ such that $y \le_\rho x$, so $y \in \mathit{SN}$. Thus $t \in \mathit{SN}$.

That is, in each case, $t \in \mathit{SN}$, and, discharging the assumptions as in the proof of Lemma 5 we get $s \in \mathit{SN}$ and $s \in \texttt{gindy} \ll (\texttt{gindy} \lhd \mathit{SN})$.

Then we prove that every term is in SN as in our Theorem 1, since both $\ll$ and $\lhd$ are well-founded. □

5. Application to Term Rewriting Systems

We now apply our Theorem 1 to the special case of a TRS on terms of a first-order language $T(\Sigma, V)$ (see [4, §3.1]), thereby showing that our main result from [8] is a special case of our Theorem 1. We consider a binary relation σ, which is the set of substitutional instances of a set of rewrite rules, and so is closed under substitutions.

However σ itself is typically not monotonic, ie, compatible with Σ-contexts (see [4, Definition 3.1.9]). So we define *ctxt* σ to be the "closure under contexts" of σ: that is, where $C[\_]$ is a context, and $(r, l) \in \sigma$, then $(C[r], C[l]) \in$ *ctxt* σ. Likewise we define *pctxt* ("*proper* context"): for $(r, l) \in \sigma$, if r and l are *proper* subterms of $C[r]$ and $C[l]$, then $(C[r], C[l]) \in$ *pctxt* σ.

In [8] we dealt with the termination of such rewrite relations. In discussing that work we will use "σ" for the relation there called "ρ", which is the set of substitutional instances of the rewrite rules. Then the rewrite relation is *ctxt* σ, which here we will call ρ. So $SN = wfp\ \rho = wfp\ (ctxt\ \sigma)$. The relation $\lhd$ of the previous sections will now be interpreted as the immediate subterm relation.

Recall that in [8] we used a relation $<_{dt} = <_{cut} \cup <_{sn1}$, where $<_{cut}$ was chosen by the user, but $<_{sn1}$ was defined as below [8, Definition 3].

Definition 6 ($<_{sn1}$, $<_{sn2}$) *Given ρ and $<_{cut}$ we define two further binary relations on terms, $<_{sn1}$ and $<_{sn2}$.*

- *$t_1 <_{sn1} t_0$ if t_0 and t_1 are the same except that an immediate subterm of t_0 is in SN and reduces to the corresponding immediate subterm of t_1.*
- *$t_1 <_{sn2} t_0$ if t_0 and t_1 are the same except that a proper subterm of t_0 is in SN and reduces to the corresponding proper subterm of t_1.*

Note that $t_1 <_{sn1} t_0$ implies $t_1 <_{sn2} t_0$, and our main theorem uses only $<_{sn1}$. However $<_{sn2}$ is sometimes easier to work with because it is closed under context.

Theorem 6 *The relations $<_{sn1}$ and $<_{sn2}$ are each well-founded [7].*

Then we apply our Theorem 1 by letting $\ll$ be the relation $<_{dt}^{+}$, which is well-founded if and only if $<_{dt}$ is so. Also let $\ll'$ be $<_{cut}$, so $\ll\ = \ \ll' \cup <_{sn1}$. We now reproduce Theorem 2 of [8] in our current notation, as Theorem 7 and Condition 2(i). Condition 2(ii) implies Condition 2(i), is more generally useful, and is used in §6.

Condition 2 *For all $(t, s) \in \sigma$,*

(i) If $\forall s' \lhd^{+} s.\ s' \in SN$ then, for all $t' \lhd^{*} t$, either $t' \in SN$ or $t' \ll s$.

(ii) For all $t' \lhd^{*} t$, either $(t', s) \in \lhd \circ (\rho \cup \lhd)^{*}$ or $t' \ll' s$.

Theorem 7 *If σ satisfies Condition 2(i) or (ii), $\ll$ contains $<_{sn1}$ and $\ll$ is well-founded, then every term is strongly normalising [8].*

Proof. We apply our Theorem 1 to this situation. Since $\lhd$ is well-founded and we assume $\ll$ is well-founded, conditions (i) and (ii) of our Theorem 1 are satisfied. It remains only to check that Condition 1(i) holds. In fact we can show that the stronger Condition 1(iv) holds.

Consider $(t, s) \in \rho$, and assume that $\forall s' \lhd s.\ s' \in SN$. As $\rho = ctxt\ \sigma$ is closed under context, it follows that any subterm of a strongly normalising term is strongly normalising, so we can assume that $\forall s' \lhd^{+} s.\ s' \in SN$. For the case $(t, s) \in \sigma$, Condition 2(i) then implies that for $t' \lhd^{*} t$, either $t' \in SN$ or $t' \ll s$, and so Condition 1(iv) holds in this case.

We also need to consider the case $(t, s) \in \rho \setminus \sigma$: that is, where a proper subterm of s is reduced, using σ, to the corresponding proper subterm of t. Consider any subterm t' of t. We show that either $t' \in SN$ or $t' <_{sn1} s$, whence $t' \ll s$.

If $t' = t$, then $t' <_{sn1} s$ by definition of $<_{sn1}$. If t' is a proper subterm of t, then if there is a corresponding subterm s' of s such that either $t' = s'$ or $(t', s') \in \mathit{ctxt}\ \sigma$, then s' and t' are in *SN*. Otherwise, there is a proper subterm s' of s, where s' is reduced to t'', so $(t'', s') \in \mathit{ctxt}\ \sigma$, and t' is a subterm of t''. Then s', t'' and t' are all in *SN*. So Condition 1(iv) holds for this case also. □

In the remainder of this section we relate this theorem to known results on constricting derivations, show how it implies known results relating to the the Knuth-Bendix Ordering, and to dependency pairs. Note that we have already discussed, in §3.2, how the termination of the recursive path orderings [12, Defn 4.22, 4.24] follows from our Theorem 1. We did in fact show, in [8, §3.7], that this also follows from Theorem 7.

We then show how the theorem can be used in a number of specific examples. WE PROPOSE TO OMIT THE TWO EXAMPLES INDICATED, FOR REASONS OF SPACE, UNLESS THE EDITORS/REFEREES HAVE DIFFERENT SUGGESTIONS. Further examples may be found in [8]. In these examples, the crux is to find an appropriate definition of $\ll'$, for this determines $\ll$ as $\ll' \cup <_{sn1}$. Since we need to choose $\ll'$ so that $\ll$ is well-founded, we choose $\ll'$ well-founded, and such that $\ll'$ and $<_{sn1}$ satisfy one of the conditions of Lemma 6. Since $<_{sn1}$ is necessarily well-founded, this ensures that $\ll$ is well-founded.

5.1. Constricting Derivations

For a rewrite system on a first-order language (where the reduction relation is closed under context), a "constricting derivation" has been defined as an infinite reduction sequence where each reduction occurs at a subterm t whose proper subterms are all strongly normalising [13].

For a rewrite relation $\rho = \mathit{ctxt}\ \sigma$, we define a *constricting reduction* by: $(t, s) \in$ *constrict* σ iff $(t, s) \in \sigma$ and the proper subterms of s are in *wfp* ρ. As before, $\lhd$ is the immediate subterm relation, and $<_{sn1}$ is defined as in §5.

The following results are easily proved by methods similar to those of [15].

Lemma 13 *For all binary relations σ and τ:*

(i) $\sigma \circ \tau$ is well-founded if and only if $\tau \circ \sigma$ is well-founded

(ii) if τ is well-founded then wfp $(\tau^* \circ \sigma) =$ wfp $(\sigma \cup \tau)$

(iii) if τ is well-founded and $\tau \circ \sigma \subseteq \sigma \circ \tau^*$, then wfp $(\sigma \cup \tau) =$ wfp σ.

The following theorem encapsulates mostly known results, for example Lemma 1 of Hirokawa & Middeldorp [20] resembles: "if (23) then (26)", and Proposition 1 of Borralleras, Ferreira & Rubio [5] is: "if (26) then (25)".

Theorem 8 *The following are equivalent, where $\rho = ctxt\ \sigma$:*

$$(constrict\ \sigma \circ \lhd^*) \cup <_{sn1} \quad \textit{is well-founded} \tag{20}$$
$$constrict\ \sigma \circ \lhd^* \circ <_{sn1}^* \quad \textit{is well-founded} \tag{21}$$
$$\lhd^* \circ <_{sn1}^* \circ constrict\ \sigma \quad \textit{is well-founded} \tag{22}$$
$$<_{sn1}^* \circ constrict\ \sigma \circ \lhd^* \quad \textit{is well-founded} \tag{23}$$
$$\rho \circ \lhd^* \quad \textit{is well-founded} \tag{24}$$
$$\rho \cup \lhd \quad \textit{is well-founded} \tag{25}$$
$$\rho \quad \textit{is well-founded} \tag{26}$$

Proof. (22) $\Rightarrow$ (26): If t_0 is not in $SN = wfp\ \rho$, then let t_0' be a minimal subterm of t_0 which is not in SN— so the proper subterms of t_0' are in SN. Consider any infinite sequence of reductions from t_0' — these cannot all be reductions of proper subterms as the latter are in SN, so find t_0'' and t_1 in this infinite sequence such that $(t_0'', t_0') \in (ctxt\ \sigma \setminus \sigma)^*$ and $(t_1, t_0'') \in \sigma$. Now all proper subterms of t_0', of t_0'' and of all terms between them in the reduction sequence are in SN. So $(t_1, t_0) \in \lhd^* \circ <_{sn1}^* \circ constrict\ \sigma$, and as $t_1 \notin SN$, a $(\lhd^* \circ <_{sn1}^* \circ constrict\ \sigma)$-sequence can be continued. Similar proofs are at, eg, [3, Theorem 6], [20, Lemma 1].

(21) $\Leftrightarrow$ (22) $\Leftrightarrow$ (23): these follow from Lemma 13(i).

(20) $\Leftrightarrow$ (23) : and (24) $\Leftrightarrow$ (25) : follows from Lemma 13(ii), since $<_{sn1}$ is well-founded.

(26) $\Rightarrow$ (25) : follows from Lemma 13(iii), as $\lhd \circ \rho \subseteq \rho \circ \lhd$, as ρ is monotonic.

(25) $\Rightarrow$ all others : since every relation mentioned is contained in $(\rho \cup \lhd)^+$. □

We now link Theorem 8 with Theorem 7, by some simple proofs. Note that Condition 2(i) in §5 simply says: if $(t', s) \in constrict\ \sigma \circ \lhd^*$ then $t' \in SN$ or $t' \ll s$.

Let $\tau = (constrict\ \sigma \circ \lhd^*) \cup <_{sn1}$. We show $SN = wfp\ \rho \subseteq wfp\ \tau$. Since $\tau \subseteq (\rho \cup \lhd)^+$, so $wfp\ (\rho \cup \lhd) \subseteq wfp\ \tau$. By Lemma 13(iii), $wfp\ \rho = wfp\ (\rho \cup \lhd)$.

Then, we can use Theorem 7 to prove (20) implies (26). Let $\ll\, = \tau$. Then Condition 2 holds trivially, $\ll\, \supseteq <_{sn1}$ and $\ll$ is well-founded, which is just (20) of Theorem 8. Hence, by Theorem 7, ρ is well-founded.

The converse is also easy to prove, giving an alternative proof of Theorem 7. Assume (20) $\Rightarrow$ (26); we will prove Theorem 7. Suppose the assumptions of Theorem 7 hold, and we want to show its conclusion, (26). We show (20), ie, that τ is well-founded. Let $(t, s) \in \tau$. Then, if $(t, s) \in <_{sn1}$, then $t \ll s$. Otherwise, $(t, s) \in constrict\ \sigma \circ \lhd^*$ and so $t \in SN$ or $t \ll s$ by Condition 2. Now $SN \subseteq wfp\ \tau$, as shown above, so $(t, s) \in \tau$ implies $t \in wfp\ \tau$ or $t \ll s$, where $\ll$ is well-founded. Thus any τ-descending chain either terminates or hits a member of $wfp\ \tau$: that is, every $s \in$ `bars` $\tau\ (wfp\ \tau)$. So, by Lemma 3(ii), every $s \in wfp\ \tau$, as required.

5.2. The Knuth-Bendix Ordering

For a rewrite system (with rules containing variables), the Knuth-Bendix ordering $<_{kb}$ is based on a strict well-founded order $<$ on function symbols and, additionally, a weight function w on function symbols and variables. Since our approach is based on a relation ρ (which amounts to the rewrite rules after all possible substitutions for variables), we describe the Knuth-Bendix ordering in this context. Weights are natural numbers, and the weight of any constant or object language variable is positive: thus every subterm has positive weight. At most one unary function symbol (call it k) can have zero weight, and then $k > f$ for any other function symbol f. The weight of a term is the sum of the weights of the function symbols and variables in it. Then we have that $s >_{kb} t$ iff $w(s) \geq w(t)$ and one of

$$w(s) > w(t) \tag{27}$$
$$s = k^n(t) \text{ for some } n > 0 \tag{28}$$
$$s = f(\overline{s}) \text{ and } t = g(\overline{t}) \text{ where } f > g \tag{29}$$
$$s = f(\overline{s}), t = f(\overline{t}) \text{ and } (\overline{t}, \overline{s}) \in \mathit{lex}\ (<_{kb}) \tag{30}$$

Theorem 9 *The Knuth-Bendix ordering is well-founded.*

Proof. We define $s \gg' t$ iff $w(s) \geq w(t)$ and either one of the rules (27) or (29) holds, or $s >_{kw1} t$, where $<_{kw1}$ is defined by:

$$f(\overline{t}) <_{kw1} f(\overline{s}) \quad \text{iff} \quad (\overline{t}, \overline{s}) \in \mathit{lex}\ (\mathit{fwf}\ (<_{kb})) \tag{31}$$

To show $\ll'$ is well-founded, we have that each of rules (27), (29) and (31) provides a well-founded relation and we can apply Lemma 6 repeatedly to show that their union is well-founded, noting that if $s >_{kw1} t$ then $w(s) \geq w(t)$. Further, $<_{sn1} \subseteq <_{kw1}$, so $\ll = \ll' \cup <_{sn1} = \ll'$ is well-founded.

To show that Condition 2(ii) is satisfied, suppose that $l >_{kb} r$ and that r' is a subterm of r. Firstly, let $r' = r$. Assume that all proper subterms of l are in *SN*, in which case, if $l >_{kb} r$ by rule (30), then $l >_{kw1} r$, so $l \gg' r$. If $l >_{kb} r$ by rules (27) or (29), then $l \gg' r$, and if $l >_{kb} r$ by rule (28) then $l \rhd^+ r$.

Now suppose r' is a proper subterm of r, so $w(r) \geq w(r')$. If $w(r') < w(r)$ or $w(r) < w(l)$ then $w(l) > w(r')$ and $l \gg' r'$ by rule (27). If $w(l) = w(r) = w(r')$ then $r = k^n(r')$ for some $n > 0$, and so $l >_{kb} r$ by either (28) or (30) (as the g of (29) cannot be k). If $l >_{kb} r$ by (28) then $l \rhd^+ r'$. If $l >_{kb} r$ by (30), then, for some l_1, r_1, we must have $l = k(l_1)$, $r = k(r_1)$, $l_1 >_{kb} r_1$ and r' is a subterm of r_1, so $(r', l) \in \lhd \circ <_{kb} \circ \lhd^*$, so satisfying Condition 2(ii). □

5.3. Dependency Pairs

Arts & Giesl [3] describe a method of establishing termination using "dependency pairs". They distinguish function symbols which appear at the head of the left-hand side of a rewrite rule ("defined symbols") and those which do not ("constructor symbols"). They follow the convention that for a rule $l \to r$ of a rewrite system, l is not a lone variable, and any variable in r is also in l. For each defined symbol d they introduce a new "tuple symbol" $d^\sharp$. From a term t we obtain a term $t^\sharp$ by changing the head symbol of t to the corresponding tuple symbol.

Previously we have considered a "rule" $l \to r$ after substitution, and the variables in our analyses have been metavariables where, for example, we might have considered a rule $g(x) \to x$ with $l = g(x)$, $r = x$, and r' a proper subterm of x.

This approach no longer holds: $l \to r$ will mean a rule before substitution, and we will use σ for a substitution.

For a rewrite rule $l \to r$, and subterm r' of r, if the head of r' is a defined symbol then $l^\sharp \to r'^\sharp$ is a *dependency pair.* In [3], the relations $\precsim$ and $\prec$ below were a quasi-order and its strict part, but in subsequent papers [17], [20] these relations needed only to be a *reduction pair*, as defined next.

Definition 7 *A* reduction pair $(\succsim, \succ)$ *consists of a quasi-ordering (ie, a reflexive and transitive relation) $\succsim$, which is closed under contexts (monotonic), and a well-founded ordering $\succ$, where both are closed under substitutions and $\succ$ is* compatible *with $\succsim$, ie either $\succsim \circ \succ \subseteq \succ$ or* [a] *$\succ \circ \succsim \subseteq \succ$.*

We now state and prove the analogue, for reduction pairs, of the "sufficiency" part of [3, Theorem 7].

Theorem 10 *If there is a reduction pair $(\succsim, \succ)$ such that*

(a) $l \succsim r$ for all rules $l \to r$, and

(b) $s \succ t$ for all dependency pairs $s \to t$

then the rewrite system terminates.

Proof. Assume that $\succsim$ and $\succ$ is minimal such that the conditions hold. Then there is no instance of $c(x) \succ d(y)$ or $c(x) \succsim d(y)$ where c is a constructor symbol and d is a defined symbol (either with or without $\sharp$), as neither (a) nor (b) nor the requirement that $\succsim$ be closed under context require it, whence the compatibility condition cannot require it.

For a substitution σ, constructor symbol c and defined symbol d, we define $\ll'$:

$$s^\sharp \prec t^\sharp \Longrightarrow s\sigma \ll' t\sigma \tag{32}$$

$$c(x) \ll' d(y) \tag{33}$$

Then $\ll'$ is well-founded because $\prec$ is and there is no instance of $c^\sharp(x) \succ d^\sharp(y)$.

To show that $\ll' \cup <_{sn1}$ is well-founded, in the case $\prec \circ \precsim \subseteq \prec$, we use Lemma 6(a), showing that $\ll' \circ <_{sn1} \subseteq \ll'$; in the case $\precsim \circ \prec \subseteq \prec$, we use Lemma 6(b), showing $<_{sn1} \circ \ll' \subseteq \ll'$.

In the case that $\prec \circ \precsim \subseteq \prec$, suppose that $t \gg' u >_{sn1} v$. If $t \gg' u$ by rule (33) then $t \gg' v$ by the same rule. If $t \gg' u$ by rule (32) then $t^\sharp \succ u^\sharp$ (since $\succ$ is closed under substitution), and a proper subterm u' of u is rewritten to a corresponding subterm v' of v. As $\precsim$ is closed under substitution, assumption (a) gives us that $u' \succsim v'$. Then, since $\succsim$ is closed under context, $u^\sharp \succsim v^\sharp$, and so $t^\sharp \succ v^\sharp$ and $t \gg' v$.

In the case that $\precsim \circ \prec \subseteq \prec$, suppose that $t >_{sn1} u \gg' v$. If $u \gg' v$ by rule (33) then $t \gg' v$ by the same rule. If $u \gg' v$ by rule (32) then by a similar argument to that above, we get $u^\sharp \succ v^\sharp$, where a proper subterm t' of t is rewritten to a corresponding subterm u' of u, and we have $t' \succsim u'$, $t^\sharp \succsim u^\sharp$, $t^\sharp \succ v^\sharp$ and $t \gg' v$.

Finally, we show that Condition 2(ii) is satisfied, For a rule $l \to r$ and substitution σ, so $(r\sigma, l\sigma) \in \rho$, and subterm r' of $r\sigma$, there are three cases for r':

- the head of r' is a constructor symbol in r, in which case $l\sigma \gg' r'$ by (33)

[a] some papers say "and" here, apparently because it makes some proofs simpler

- the head of r' is a defined symbol in r, in which case $r' = r_1\sigma$ for some subterm r_1 of r, $l^\sharp \to r_1^\sharp$ is a dependency pair, and $l\sigma \gg' r_1\sigma = r'$ by (32)
- r' is a subterm of $x\sigma$ for some variable x in r, in which case r' is a proper subterm of $l\sigma$, since any variable in r appears as a proper subterm of l. □

5.4. Multiset order : PROPOSED TO BE OMITTED

Given an irreflexive relation ρ on a set E, we can define the *multiset order* derived from ρ on multisets of elements of E. We use A, B and C for finite multisets of elements of E, by which we mean both that they contain only finitely many distinct elements and that they contain only finitely many copies of each such element. We use $A \sqcup B$ to stand for the multiset union. We consider the irreflexive relation $<_{m1}$ defined on finite multisets:

$<_{m1}$**:** $\forall C, \forall b \in E$. if, for all $c \in C$, $c <_\rho b$, then $C \sqcup A <_{m1} \{b\} \sqcup A$.

If ρ is a strict order (an irreflexive, transitive relation), then $<_{m1}^+$ is equal to the multiset order derived from ρ.

We represent a multiset as a tree, with two sorts of node, "inner" nodes (I) and "leaf" nodes (L). Viewing such a tree as a term, the function symbols are I and $L(e)$ for each $e \in E$, where I has arbitrary arity and each $L(e)$ is nullary. The "leaf multiset" of a tree is the multiset of its leaf nodes, but with each $L(e)$ changed to e. Note that different trees can have the same leaf multiset.

We define a rewrite relation on such trees (terms) as follows. For every (finite) multiset $C = [c_1, c_2, \ldots, c_k]$ and every element $b \in E$ satisfying $\forall c_i \in C.\ c_i <_\rho b$ (as in the definition of $<_{m1}$) we have a rule $L(b) \longrightarrow I(L(c_1), L(c_2), \ldots, L(c_k))$.

Theorem 11 *Given a well-founded order ρ, the derived multiset order (on finite multisets) is well-founded.*

Proof. Clearly, whenever $A <_{m1} B$, any tree whose leaf multiset is B can be reduced to a tree whose leaf multiset is A using the rewrite relation defined above. Since for any finite multiset B there is a tree whose leaf multiset is B, when we show that rewriting terminates we have shown that the multiset order is well-founded.

We prove that this rewrite system terminates. We define $\ll'$ by the rules

$$L(x) \gg' I(y) \qquad\qquad L(x) \gg' L(y) \text{ iff } x >_\rho y$$

It is clear that $\ll'$ is well-founded when ρ is. To show that $\ll' \cup <_{sn1}$ is well-founded, we use Lemma 6, by showing that in fact $<_{sn1} \circ \ll' = \emptyset$. For suppose $t >_{sn1} u \gg' v$. Then u must be of the form $L(x)$, and a proper subterm of t must reduce to a proper subterm of u – but u has no proper subterms.

To show that Condition 2 is satisfied, when $L(b) \longrightarrow I(L(c_1), \ldots, L(c_k))$, we have $L(b) \gg' I(L(c_1), \ldots, L(c_k))$ by the first rule for $\ll'$, and, for every subterm $L(c_i)$ of the reduced subterm, $L(b) \gg' L(c_i)$ by the second rule. □

5.5. A non simplification ordering

Example 5 of [11], with the single rule $f(f(x)) \longrightarrow f(g(f(x)))$ is one for which a simplification ordering cannot be used, because a simplification ordering would take $g(f(x))$ to $f(x)$ and so $f(g(f(x)))$ to $f(f(x))$, giving a cycle.

But Theorem 7 is not limited to simplification orderings. We define $\ll'$ according to the number of consecutive f symbols starting from the head of a term. Alternatively, we could use the total number of pairs of adjacent f symbols, as suggested in [11]. Thus $f(f(x)) \gg' f(g(y))$, $f(f(x)) \gg' g(y)$, and $f(f(x)) \gg' f(x)$. Finally, any subterm of x is a proper subterm of $f(f(x))$. Thus Condition 2(ii) is satisfied. Clearly also, rewriting a subterm cannot increase the number of consecutive f symbols, so $<_{sn1} \circ \ll' \subseteq \ll'$ (and likewise $\ll' \circ <_{sn1} \subseteq \ll'$). Thus $\ll \, = \, \ll' \cup <_{sn1}$ is well-founded by Lemma 6. Therefore the system terminates, by Theorem 7.

5.6. Ackermann's function : PROPOSED TO BE OMITTED

Ackermann's function on the natural numbers can be defined by the following rewrite rules [11, Example 29]

$$\begin{aligned} A(0, y) &\longrightarrow S(y) \\ A(S(x), 0) &\longrightarrow A(x, S(0)) \\ A(S(x), S(y)) &\longrightarrow A(x, A(S(x), y)) \end{aligned}$$

It can be shown to terminate using the lexicographic path ordering. This is reflected in the relation $\gg'$ we use, which is defined by the following cases:

$$A(x, y) \gg' S(z) \tag{34}$$
$$A(S(x), y) \gg' A(x, z) \tag{35}$$
$$A(x, S(y)) \gg' A(x, y) \tag{36}$$

We now prove that this rewrite system terminates. It is clear that $\gg'$ is well-founded using the lexicographic ordering on arguments. It is also clear that for each (r', l), where $l \to r$ is a rewrite rule and r' is a subterm of r, either $l \gg' r'$ or r' is a proper subterm of l.

It remains to show that $\ll' \cup <_{sn1}$ is well-founded. Again we show that $\ll' \cup <_{sn2}$ is well-founded, using Lemma 6(a). We show that $\ll' \circ <_{sn2} \subseteq <^*_{sn2} \circ \ll'$. For suppose $t \gg' u >_{sn2} v$. If $t \gg' u$ by rule (34), ie $t = A(x, y)$ and $u = S(z)$, then $v = S(z')$ and so $t \gg' v$. If $t \gg' u$ by rule (35), then $t = A(S(x), y)$ and $u = A(x, z)$. There are two cases for $u >_{sn2} v$: $v = A(x, z')$ where $z \longrightarrow z'$, in which case $t \gg' v$, or $v = A(x', y)$ where $x \longrightarrow x'$ by reducing a strongly normalising subterm of x, in which case $t = A(S(x), y) >_{sn2} A(S(x'), y) \gg' A(x', z) = v$. The case for rule (36) is similar.

So in all cases $(v, t) \in <^*_{sn2} \circ \ll'$, and so $\ll' \cup <_{sn2}$ is well-founded, by Lemma 6(a). Therefore the system terminates by Theorem 7.

5.7. The factorial example

Example 21 of [11] is almost the usual definition of the factorial function, but modified so that we can not use a simplification ordering. The rules are

$$\begin{array}{rcl} P(S(x)) & \longrightarrow & x \\ F(S(x)) & \longrightarrow & S(x) \times F(P(S(x))) \\ S(x) \times y & \longrightarrow & x \times y + y \\ x + S(y) & \longrightarrow & S(x+y) \end{array} \qquad \begin{array}{rcl} F(0) & \longrightarrow & 0 \\ 0 \times y & \longrightarrow & 0 \\ x + 0 & \longrightarrow & x \end{array}$$

As usual we define a (transitive) ordering $<_h$ of terms based on the following ordering of head symbol: $F > \times > + > S$. But we need to define $\ll'$ to be the union of $<_h$ and the following additional cases

$$F(S(x)) \gg' F(P(S(x))) \qquad F(S(x)) \gg' F(P(x))$$

We can not use a simplification ordering because if we allowed $P(x) \longrightarrow x$ the system would not terminate, but would "cycle" between terms containing $F(S(x))$ and terms containing $F(P(S(x)))$. We do, however, need to add the rule $S(x) \longrightarrow x$. The proofs of termination given in [11, Examples 21, 25] are based on interpreting arguments to the function symbols as natural numbers.

Now for each (r', l) where $l \to r$ is a rewrite rule and r' is a subterm of r, it is reasonably easy to see that we have one of the following cases:

- r' is a proper subterm of l
- $r' <_h l$
- r' is l, but with S removed from a subterm (to give $r' <_{sn1} l$)
- r' is $F(P(S(x)))$ and l is $F(S(x))$ (so $r' \ll' l$).

It is easy to see that $\ll'$ is well-founded. To show that $\ll' \cup <_{sn2}$ is well-founded we need to use a more general case of Lemma 6 than hitherto. In fact we show $\ll' \circ <_{sn2} \subseteq (<^*_{sn2} \circ \ll') \cup <^+_{sn2}$, which implies (d) of Lemma 6.

Suppose $t \gg' u >_{sn2} v$. If $t >_h u$ then clearly $t >_h v$. Suppose $t = F(S(x))$ and $u = F(P(S(x)))$ or $u = F(P(x))$. If $u >_{sn2} v$ by way of reducing the x in u to x' (reducing a strongly normalising subterm of x), then $t = F(S(x)) >_{sn2} F(S(x')) \gg' v$. Otherwise we need to consider specific cases:

$u = F(P(S(x)))$, $S(x) \longrightarrow x$ where $S(x) \in SN$, $v = F(P(x))$:
in this case, $t \gg' v$

$u = F(P(S(x)))$, $P(S(x)) \longrightarrow x$ where $P(S(x)) \in SN$, $v = F(x)$:
as $S(x) \longrightarrow x$ and $S(x) \in SN$, we have $t >_{sn2} v$

$u = F(P(x))$, where $x = S(y)$, $P(S(y)) \longrightarrow y$ and $P(S(y)) \in SN$, $v = F(y)$:
as $S(y) \longrightarrow y$ and $S(y) \in SN$, so $t = F(S(S(y)) >_{sn2} F(S(y)) >_{sn2} F(y) = v$.

Thus $\ll' \cup <_{sn2}$ is well-founded and the system terminates by Theorem 7.

6. Incremental Proofs of Termination

A rewrite system can be defined by taking the union of two terminating systems. Obviously, it would be desirable to be able to reduce a proof of termination of such a system into two smaller proofs of termination of smaller systems. This is possible under certain conditions (see, eg, [16]), but not in general. We show how [8, Theorem 2], can be used to prove termination incrementally in certain cases.

The assumptions we require of the component systems are mentioned where they first become relevant, and all appear in Theorem 12.

Let $\mathcal{R}_0$ be a set of rewrite rules, in a first-order language, where the function symbols appearing in the rules are from the set $\mathcal{F}_0$. Note, however, that the variables appearing in those rules may be replaced by any term (which may contain function symbols outside $\mathcal{F}_0$). The set of substitution instances of the rules in $\mathcal{R}_0$ is the relation σ_0, with corresponding rewrite relation $\rho_0 = ctxt\ \sigma_0$.

We consider a rewrite system ρ_0 which has been proved terminating by any method, with only the extra condition that the rules $\mathcal{R}_0$ be *right-linear*: that is, no variable appears more than once on the right-hand side of a rule. Thus we define the $\mathcal{R}_0$*-property*, (in which (ii) and (iv) are obviously necessary for any terminating system), and assume throughout that $\mathcal{R}_0$ satisfies it. The $\mathcal{R}_0$*-property* is important for the lemma which follows.

Definition 8 ($\mathcal{R}_0$-*property*) *A rule satisfies the $\mathcal{R}_0$-property if*

(i) its function symbols are in the set $\mathcal{F}_0$
(ii) its left-hand side is not a variable
(iii) its right-hand side variables are not duplicated
(iv) its right-hand side variables also appear on the left-hand side.

Lemma 14 *Let σ be the set of substitution instances of a set of rules satisfying the $\mathcal{R}_0$-property. Then*

(i) For $f \notin \mathcal{F}_0$ and $(t, s) \in \sigma$, if $t' = f(\overline{t}) \lhd^* t$ then $t' \lhd^+ s$.
(ii) Let τ be a relation such that for $(t, f(\overline{t})) \in \tau$, $f \notin \mathcal{F}_0$. Then $\sigma \circ pctxt\ \tau \subseteq (pctxt\ \tau)^* \circ \sigma$.

We then consider a second rewrite system $\rho_1 = ctxt\ \sigma_1$ whose "defined symbols" are from a set of new symbols $\mathcal{F}_1$, where $\mathcal{F}_0 \cap \mathcal{F}_1 = \emptyset$. That is, for $(t, s) \in \sigma_1$, s is of the form $f(\overline{s})$, for some $f \in \mathcal{F}_1$ and some sequence $\overline{s}$ of terms. The system ρ_1 is assumed to have been proved terminating using Theorem 7 above (ie, using [8, Theorem 2]): that is, by defining a relation $\ll'_1$ such that $\ll_1 = \ll'_1 \cup <_{sn1}$ is well-founded, where $<_{sn1}$ is defined in terms of ρ_1 only.

In many examples of the use of Theorem 7, the argument went as follows. We let $<_{sn2}$ be the set of those reductions where a strongly normalising *proper* subterm is reduced, so $<_{sn1} \subseteq <_{sn2}$, and $<_{sn2}$ is also necessarily well-founded, by Theorem 6. Then, using $<_{sn2}$ in place of $<_{sn1}$, we prove that $\ll' \cup <_{sn2}$ is well-founded by proving that $\ll'$ is well-founded and then using Lemma 6 to prove that the union is well-founded, often by proving that $\ll' \circ <_{sn2} \subseteq\ {<_{sn2}}^* \circ \ll'$.

We use this proof method. The key is to define a suitable relation $\ll'$: we will use $\ll' = \ll'_0 \cup \ll'_1$, where $\ll'_0$ is a suitable relation which we derive from $\mathcal{R}_0$. To help prove $\ll' \circ <_{sn2} \subseteq\ {<_{sn2}}^* \circ \ll'$ we will assume $\ll'_1$ satisfies the $\ll'_1$*-property*:

Definition 9 ($\ll'_1$-*property*) *A relation $\ll'_1$ satisfies the $\ll'_1$-property if, for* any *relation σ, $\ll'_1 \circ pctxt\ \sigma \subseteq (pctxt\ \sigma)^* \circ \ll'_1$.*

In fact, the $\ll'_1$-property could be weakened, to apply only to certain choices of σ, and to match (d) instead of (a) of Lemma 6, and Theorem 12 still holds. The example in [8, §3.6] satisfies only the weaker condition.

To define $\ll'_0$ we first define $\lhd_0$ and $ctxt_0$, and then a set of rules $\mathcal{R}_{\ll 0}$:

Definition 10 **($\lhd_0$)** *$t \lhd_0 f(\ldots, t, \ldots)$ if $f \in \mathcal{F}_0$.*

Definition 11 **($ctxt_0$)** *For $(t, s) \in \sigma$, if the subterms of $C[x]$ which contain x have head symbols in $\mathcal{F}_0$ then $(C[t], C[s]) \in ctxt_0\ \sigma$.*

Definition 12 **($\mathcal{R}_{\ll 0}$)** *$(r'_0, l_0) \in \mathcal{R}_{\ll 0}$ iff it satisfies the $\mathcal{R}_0$-property and there exists r_0 such that $r'_0 \lhd_0^* r_0$ and r'_0 is not a variable, and $(r_0, l_0) \in ctxt_0\ \mathcal{R}_0$.*

Then let $\ll'_0$ be the set of substitution instances of the rules in $\mathcal{R}_{\ll 0}$. Clearly $\ll'_0 \subseteq (\rho_0 \cup \lhd)^+$ so $\ll'_0$ is well-founded.

We now can define $\ll' = \ll'_0 \cup \ll'_1$. If we assume that for $t \ll'_1 f(\overline{t})$, $f \in \mathcal{F}_1$, and that $\ll'_1$ is well-founded, then $\ll'$ is well-founded as $\ll'_0 \circ \ll'_1 = \emptyset$. Then define $\ll\ = \ll' \cup <_{sn2}$. Since $<_{sn2}$ is well-founded (Theorem 6), to show that $\ll$ is well-founded it would be enough to show $\ll' \circ <_{sn2} \subseteq <_{sn2}{}^* \circ \ll'$. We cannot do this, but we can choose a suitable subset $<'_{sn2}$ of $<_{sn2}$ such that $\ll\ = \ll' \cup <'_{sn2}$ and we prove that $\ll$ is well-founded in two steps.

We define τ_1: $(t, f(\overline{t})) \in \tau_1$ iff $f \notin \mathcal{F}_0$. Recall that SN means the set of strongly normalising terms, ie $SN = wfp\ (ctxt\ \rho)$. We define ρ_{SN}: $(t, s) \in \rho_{SN}$ iff $s \in SN$. Thus $<_{sn2} =\ pctxt\ (\sigma \cap \rho_{SN})$. Then the required relation $<'_{sn2}$ is given in the following lemma, whose proof is detailed but tedious, and which has been proved in Isabelle ([10], file `hier.ML`), as have all these results.

Lemma 15 *Let $<'_{sn2} = ctxt\ (\tau_1 \cap pctxt\ (\sigma_0 \cap \rho_{SN})) \cup pctxt\ (\sigma_1 \cap \rho_{SN})$ Let $\mathcal{R}_0$ satisfy the* $\mathcal{R}_0$-property*. Then $<'_{sn2} \subseteq <_{sn2} \subseteq \ll'_0 \cup <'_{sn2}$.*

Lemma 16 *Assume hypotheses (i), (ii) and (iv) of Theorem 12 below. Then*

- *(i) $\ll'_1 \cup <_{sn2}$ is well-founded; so $\ll'_1 \cup <'_{sn2}$ is well-founded.*
- *(ii) $\ll'_0 \circ \ll'_1 = \emptyset$ and $\ll'_0 \circ <'_{sn2} \subseteq <'_{sn2}{}^* \circ \ll'_0$.*
- *(iii) $\ll'_0 \cup (\ll'_1 \cup <'_{sn2})$, that is, $\ll$, is well-founded.*

Proof.

- (i) This follows from Lemma 6(a) as $\ll'_1$ satisfies the $\ll'_1$-property.
- (ii) Recall that, for $f(\overline{t}) \ll'_0 s$, $f \in \mathcal{F}_0$. Then use Lemma 14(ii).
- (iii) As $\ll'_0$ and, by (i), $\ll'_1 \cup <'_{sn2}$ are well-founded, we can use (ii) and Lemma 6(a) to get that $\ll'_0 \cup (\ll'_1 \cup <'_{sn2})$ is well-founded. Then, by Lemma 15, this is $\ll'_0 \cup (\ll'_1 \cup <_{sn2}) = \ll' \cup <_{sn2} = \ll$. □

Lemma 17 *If $\mathcal{R}_0$ satisfies the* $\mathcal{R}_0$-property*, σ_0 and $\ll'_0$ satisfy Condition 2(ii).*

Proof. Let $(t, s) \in \sigma_0$, an instance of the rule $(t_0, s_0) \in \mathcal{R}_0$, and let $t' \lhd^* t$. If $t' \lhd_0^* t$, then $(t', s) \in \ll'_0$. Otherwise, there must be $f \notin \mathcal{F}_0$ and a sequence $\overline{t}$ of terms with $t' \lhd^* f(\overline{t}) \lhd^* t$, and so, by Lemma 14(i), $t' \lhd^* f(\overline{t}) \lhd^+ s$, as required.□

Theorem 12 *Assume that*

- *(i) rules $\mathcal{R}_0$ satisfy the $\mathcal{R}_0$-property, and give a terminating rewrite system ρ_0,*
- *(ii) relation $\ll'_1$ is well-founded and satisfies the $\ll'_1$-property,*
- *(iii) σ_1 and $\ll'_1$ satisfy Condition 2(ii),*
- *(iv) for $(t, s) \in \sigma_1 \cup \ll'_1$, s is of the form $f(\overline{s})$, with $f \notin \mathcal{F}_0$.*

Then $\rho_0 \cup \rho_1$ is well-founded.

Proof. From assumption (iii) and Lemma 17, we can see that $\sigma_0 \cup \sigma_1$ and $\ll'_0 \cup \ll'_1$ satisfy Condition 2(ii). Then, since, by Lemma 16(iii), $\ll \, = \, \ll'_0 \cup \ll'_1 \cup <_{sn2}$ is well-founded, and $<_{sn1} \subseteq <_{sn2}$, the result follows from Theorem 7. □

7. An Incremental Path Ordering

We now use the previous results to describe a generalisation to an incrementally defined ordering of the general path ordering of Dershowitz & Hoot [14].

The incremental path ordering $<_{ipo}$ (or *ipo*) is then defined as below, where $\overline{s} = s_1, \ldots, s_m$, $\overline{t} = t_1, \ldots, t_n$, and $s = f(\overline{s})$ and $t = g(\overline{t})$. Let $\Lambda(ipo)$ (or $<_\Lambda$) be an ordering on lists of terms, derived from $<_{ipo}$, satisfying certain conditions given later: the common examples for Λ are the lexicographic or multiset extensions of $<_{ipo}$. As before we have a set of rules $\mathcal{R}_0$ satisfying the the $\mathcal{R}_0$-property. Again, let σ_0 be the set of substitution instances of the rules in $\mathcal{R}_0$, and let the corresponding rewrite relation $\rho_0 = ctxt\ \sigma_0$ be well-founded. Let $<$ be a well-founded ordering on the function symbols.

$$\frac{f \notin \mathcal{F}_0 \qquad s_i \geq_{ipo} t}{s >_{ipo} t} \tag{37}$$

$$\frac{f = g \quad \overline{s} >_\Lambda \overline{t} \quad f \notin \mathcal{F}_0 \quad \forall i \in \{1, \ldots, n\}.\ s >_{ipo} t_i}{s >_{ipo} t} \tag{38}$$

$$\frac{f > g \quad f \notin \mathcal{F}_0 \quad \forall i \in \{1, \ldots, n\}.\ s >_{ipo} t_i}{s >_{ipo} t} \tag{39}$$

$$\frac{(t, s) \in \sigma_0}{s >_{ipo} t} \tag{40}$$

$$\frac{(t, s) \in pctxt\ ipo}{s >_{ipo} t} \tag{41}$$

Rules (41) and (40) imply $\rho_0 = ctxt\ \sigma_0 \ \subseteq <_{ipo}$ and that *ipo* is closed under contexts. In the Isabelle formulation *ipo* is an inductively defined set, where *ipo* is the set of all pairs whose inclusion in *ipo* is established by the rules given.

Note that if $\mathcal{F}_0$, and so $\mathcal{R}_0$ and σ_0 are empty, then this reduces to the recursive path ordering. In that case, if Λ is the lexicographic or multiset ordering, then rule (38) implies (41). Also, in that case, the rules themselves imply that the defined path ordering is transitive, and this fact is used in some proofs of well-foundedness (see, eg, [14]). But when $\mathcal{F}_0$, $\mathcal{R}_0$ and σ_0 are non-empty, it does not seem clear whether $<_{ipo}$ is transitive, and our proof of termination does not depend on it.

As in [8], we define a function *fwf* ("from well-founded") which maps a binary relation σ to a binary relation $fwf\ \sigma$ thus: $(t, s) \in fwf\ \sigma$ iff $(t, s) \in \sigma$ and $s \in wfp\ \sigma$.

We now mention the conditions on Λ which we found were required at some point in the proof.

(a) Λ is a monotonic function
(b) if σ is well-founded then $\Lambda(\sigma)$ is well-founded
(c) if all s_i are in $wfp\ \tau$ and if $(t, s) \in \Lambda(\tau)$, then $(t, s) \in \Lambda(fwf\ \tau)$

(d) if $(s_i', s_i) \in \tau$ then $((s_1, \ldots, s_i', \ldots s_m), (s_1, \ldots, s_i, \ldots s_m) \in \Lambda(\tau)$

Related conditions are discussed in §3.2, and are all satisfied by the lexicographic and multiset extensions of an ordering.

The proof of termination consists mostly of combining the proof of Theorem 12 above with ideas from the proof of the termination of the recursive path ordering in [8, §3.7]. We omit the details, but note that it has been proved using the Isabelle theorem prover, see [10], in `snabs/ipodef.{thy,ML}`.

Theorem 13 *Assume that*

(i) rules $\mathcal{R}_0$ satisfy the $\mathcal{R}_0$-property, and give a terminating rewrite system $\rho_0 = ctxt\ \sigma_0$,

(ii) the relation $<$ on the function symbols (see rule (39)) is well-founded, and

(iii) the ordering extension function Λ satisfies the conditions listed above

Then $<_{ipo}$ is well-founded.

8. Observations and Conclusion

We have proved a theorem about termination of reduction rules which generalises the previous quite general theorems, in [8] and the first theorem in [19].

We use our main theorem to prove the termination of the reduction of well-typed combinator expressions. One of our proofs takes advantage of the abstract setting, using a relation $\lhd$ which is *not* the usual immediate subterm relation.

The picture that emerges is the following. JGL1, that is, Goubault-Larrecq's Theorem 1 [19], can handle ARSs but cannot handle reducibility arguments like those required for combinators or the simply-typed λ-calculus. Our Theorem 7, that is, [8, Theorem 2], handles TRSs but also cannot handle such reducibility arguments. Our new Theorem 1 handles both TRSs and some reducibility arguments such as those involved in the proofs of termination of the typed combinator systems. However it had to be modified to reason indirectly about substitutions for the simply-typed λ-calculus. An important goal is to find the exact relationship between [19, Theorem 2] and our Theorem 1.

Finally, we show how our main theorem can be used to prove termination of a rewrite system defined incrementally. We showed in [8] that our main theorem strictly subsumes the termination of the recursive path ordering. Since [16] contains a very general set of results which are nonetheless based on the recursive path ordering, neither it nor the work in §6 subsume each other. A goal of further work is to explore the relationship between their results and the work in §7.

Acknowledgements

We wish to thank Linda Buisman for investigating some of the examples, and researching the criteria for well-foundedness of a union of well-founded orderings. We also wish to thank Jean Goubault-Larrecq and Hubert Comon for pointers to the literature. Finally, we thank some anonymous referees for very helpful comments.

References

1. Yohji Akama. A λ-to-CL Translation for Strong Normalization. In Proc. Typed Lambda Calculi and Applications (TLCA'97), LNCS 1210, 1–10, Springer, 1997.
2. Thorsten Altenkirch and James Chapman. Tait in one big step. Mathematically Structured Functional Programming (MSFP 2006). Electronic Workshops in Computing, British Computer Society, 2006.
3. Thomas Arts and Jürgen Giesl. Termination of Term Rewriting Using Dependency Pairs. Theoretical Computer Science 236 (2000), 133-178, 2000.
4. Franz Baader and Tobias Nipkow. Term Rewriting and All That. CUP, 1998
5. C Borralleras, M Ferreira and A Rubio. Complete Monotonic Semantic Path Orderings. In Proc. CADE-17, LNCS 1831, 346–364, Springer, 2000.
6. Wilfried Buchholz. Proof-theoretic analysis of termination proofs. Annals of Pure and Applied Logic 75 (1995), 57–65.
7. Jeremy E. Dawson and Rajeev Goré. A New Machine-checked Proof of Strong Normalisation for Display Logic. In Computing: The Australasian Theory Symposium, 2003, Electronic Notes in Theoretical Computer Science 78, 16–35, Elsevier.
8. Jeremy E Dawson and Rajeev Goré. A General Theorem on Termination of Rewriting. In Proc. Computer Science Logic (CSL 2004), LNCS 3210, 100–114, 2004.
9. J E Dawson and R Goré. Machine-checked Termination Proofs for Abstract Reduction Systems. In preparation.
10. J E Dawson. Isabelle code, `http://web.rsise.anu.edu.au/~jeremy/isabelle`
11. Nachum Dershowitz. 33 Examples of Termination. Proc. French Spring School of Theoretical Computer Science (1993), LNCS 909, 16–25, Springer 1995.
12. Nachum Dershowitz & David A. Plaisted. Rewriting. In Alan Robinson & Andrei Voronkov, eds, Handbook of Automated Reasoning, 535–610, Elsevier, 2001.
13. N Dershowitz. Termination Dependencies (Extended Abstract). In Proc. Sixth Int'l Workshop on Termination (WST '03), A Rubio, ed., Valencia, Spain, pp. 27–30.
14. N Dershowitz & C Hoot. Natural Termination. TCS 142 (1995), 179-207.
15. Henk Doornbos & Burghard Von Karger. On the Union of Well-Founded Relations. L. J. of the IGPL 6 (1998), 195-201.
16. Mirtha-Lina Fernández, Guillem Godoy, Albert Rubio. Recursive Path Orderings Can Also Be Incremental. Proc. LPAR2005, LNCS 3835, 230–245, Springer, 2005.
17. J Giesl, T Arts, and E Ohlebusch. Modular Termination Proofs for Rewriting Using Dependency Pairs. Journal of Symbolic Computation 34(1):21-58, 2002.
18. Jean-Yves Girard, Yves Lafont and Paul Taylor. Proofs and Types. CUP, 1990.
19. J Goubault-Larrecq. Well-founded recursive relations. In Proc. CSL'2001, LNCS 2142, 484–497, Springer, 2001.
20. Nao Hirokawa and Aart Middeldorp. Dependency Pairs Revisited. In Proc. Rewriting Techniques and Applications (RTA-04), LNCS 3091, 249–268, Springer, 2004.