Iterative Lexicographic Path Orders

Jan Willem Klop[1,2,3], Vincent van Oostrom[4], and Roel de Vrijer[1]

[1] Vrije Universiteit, Department of Theoretical Computer Science,
De Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands
[2] Radboud Universiteit Nijmegen, Department of Computer Science,
Toernooiveld 1, 6525 ED Nijmegen, The Netherlands
[3] CWI, P.O. Box 94079, 1090 GB Amsterdam, The Netherlands
[4] Universiteit Utrecht, Department of Philosophy,
Heidelberglaan 8, 3584 CS Utrecht, The Netherlands
jwk@cs.vu.nl, Vincent.vanOostrom@phil.uu.nl, rdv@cs.vu.nl

Abstract. We relate Kamin and Lévy's original presentation of *lexicographic path orders* (LPO), using an inductive definition, to a presentation, which we will refer to as *iterative lexicographic path orders* (ILPO), based on Bergstra and Klop's definition of recursive path orders by way of an auxiliary term rewriting sytem.

Dedicated to Joseph Goguen, in celebration of his 65th birthday.

1 Introduction

In his seminal paper [1], Dershowitz introduced the recursive path order (RPO) method to prove termination of a first-order term rewrite system (TRS) $\mathcal{T}$. The method is based on lifting a well-quasi-order $\preceq$ on the signature of a TRS to a well-quasi-order $\preceq_{rpo}$ on the set of terms over the signature [2]. Termination of the TRS follows if $l \succ_{rpo} r$ holds for every rule $l \to r$ of $\mathcal{T}$.

In Bergstra and Klop [3] an alternative definition of RPO is put forward, which we call the *iterative* path order (IPO), the name stressing the way it is generated—see also Bergstra, Klop and Middeldorp [4]. It is operational in the sense that it is itself defined by means of an (auxiliary) term rewrite system $\mathcal{L}\mathsf{ex}$, the rules of which depend (only) on the given well-quasi-order $\preceq$.

What has been lacking until now is an understanding of the exact relationship between the recursive and iterative approaches to path orders. This will be the main subject of our investigation here. We show that both approaches coincide in the case of *transitive* relations (orders). Moreover, we provide a direct proof of termination for the iterative path order starting from an *arbitrary* terminating relation on the signature, employing a proof technique due to Buchholz [5]. Both proofs essentially rely on a natural-number-labelled variant $\mathcal{L}\mathsf{ex}^{\omega}$ of the auxiliary TRS $\mathcal{L}\mathsf{ex}$, introduced here for the first time.

For the sake of exposition we focus on the restriction of RPO due to Kamin and Lévy [6] known as the lexicographic path order (LPO)—see also Baader and

Nipkow [7]. Restricting the iterative approach accordingly gives rise to what we call the iterative lexicographic path order (ILPO), as formulated for the first time in the PhD thesis of Geser [8] and, in a slightly restricted form, by Klop [9]. As far as we know also in this case the correspondence between both has not been investigated in the literature.

The proofs that the iterative lexicographic path order is terminating and that LPO and ILPO coincide will constitute the body of the paper. In the conclusions, we put forward some ideas on the robustness of this correspondence, i.e. whether variations on LPO can be matched by corresponding variations on ILPO restoring their coincidence.

Acknowledgement We thank Alfons Geser for useful remarks.

2 The iterative lexicographic path order

The iterative lexicographic path order (ILPO) is a method to prove termination of a term rewrite system (TRS). Here a TRS $\mathcal{T}$ is *terminating* if its rewrite relation $\to_{\mathcal{T}}$ is so, i.e. if it does not allow an infinite reduction $t_0 \to_{\mathcal{T}} t_1 \to_{\mathcal{T}} t_2 \to_{\mathcal{T}} \cdots$. The method is based on iteratively lifting a terminating relation R on the signature to a terminating relation R_{ilpo} on the terms over the signature. The lifting is iterative in the sense that R_{ilpo} is defined via the *iteration* of reduction steps in an *atomic decomposition* TRS $\mathcal{L}\mathsf{ex}$ (depending on R), instead of *recursively* as in Dershowitz's recursive path order method [1]. By its definition via the atomic decomposition TRS, transitivity and closure under contexts and substitutions of R_{ilpo} are automatic, which combined with termination yields that R_{ilpo} is a so-called *reduction order* [10, Definition 6.1.2]. Therefore, for the TRS $\mathcal{T}$ to be terminating it suffices that $l\ R_{ilpo}\ r$ holds for each rule $l \to r$ in $\mathcal{T}$.

As a running example to illustrate ILPO, we take the terminating relation R given by $\mathtt{M}\ R\ \mathtt{A}$ and $\mathtt{A}\ R\ \mathtt{S}$ on the signature of the TRS $\mathcal{D}\mathsf{ed}$ of addition and multiplication on natural numbers with the rewrite rules of Table 1, going back to at least Dedekind [11].[5]

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

Table 1. Dedekind's rules for addition and multiplication

Clearly, the relation R is terminating and ILPO will lift it to a terminating relation R_{ilpo} such that $l\ R_{ilpo}\ r$ holds for each rule $l \to r$ in $\mathcal{D}\mathsf{ed}$, implying termination of $\mathcal{D}\mathsf{ed}$.

[5] Dedekind took 1 instead of 0 for the base case.

For a given relation R over the signature, the definition of R_{ilpo} proceeds in two steps: We first define the atomic decomposition TRS $\mathcal{L}\mathsf{ex}$ depending on R, over the signature extended with *control* symbols. Next, the relation R_{ilpo} is defined by restricting iteration of $\mathcal{L}\mathsf{ex}$-reduction steps, i.e. of $\to^{+}_{\mathcal{L}\mathsf{ex}}$, to terms over the original signature.

Definition 1. *Let R be a relation on a signature Σ, and let V be a signature of nullary symbols disjoint from Σ (called variables). The* atomic decomposition *TRS $\mathcal{L}\mathsf{ex}$ is $\langle \Sigma \uplus \Sigma^* \uplus V, \mathcal{R}\rangle$, where:*

1. *The signature Σ^* of* control *symbols is a copy of Σ, i.e. for each function symbol $f \in \Sigma$, Σ^* contains a fresh symbol f^* having the same arity f has.*
2. *The rules $\mathcal{R}$ are given in Table 2, for arbitrary function symbols f, g in Σ, with $\boldsymbol{x}$, $\boldsymbol{y}$, $\boldsymbol{z}$ disjoint vectors of pairwise disjoint variables of appropriate lengths.*

$$
\begin{array}{rlll}
f(\boldsymbol{x}) & \to_{\mathrm{put}} & f^*(\boldsymbol{x}) & \\
f^*(\boldsymbol{x}) & \to_{\mathrm{select}} & x_i \quad (1 \le i \le |\boldsymbol{x}|) & \\
f^*(\boldsymbol{x}) & \to_{\mathrm{copy}} & g(f^*(\boldsymbol{x}), \ldots, f^*(\boldsymbol{x})) & (f\ R\ g) \\
f^*(\boldsymbol{x}, g(\boldsymbol{y}), \boldsymbol{z}) & \to_{\mathrm{lex}} & f(\boldsymbol{x}, g^*(\boldsymbol{y}), l, \ldots, l) & (l = f^*(\boldsymbol{x}, g(\boldsymbol{y}), \boldsymbol{z}))
\end{array}
$$

Table 2. The rules of the atomic decomposition TRS $\mathcal{L}\mathsf{ex}$

The idea of the atomic decomposition $\mathcal{L}\mathsf{ex}$ is that marking the head symbol of a term, by means of the put-rule, corresponds to the obligation to make that term *smaller*, whereas the other rules correspond to *atomic* ways in which this can be brought about:

1. The select-rule expresses that selecting one of the arguments of a term makes it smaller.
2. The copy-rule expresses that a term t can be made smaller by putting copies of terms smaller than t below a head symbol g which is less heavy than the head symbol f of t.
3. The lex-rule expresses that a term t can be made smaller by making one of its subterms smaller. At the same time one may replace all the subterms to the *right* (whence the name lex) of this subterm by arbitrary terms that are smaller than the whole term t.

For our running example $\mathcal{D}\mathsf{ed}$, the reduction $\mathtt{A}(x, \mathtt{0}) \to_{\mathrm{put}} \mathtt{A}^*(x, \mathtt{0}) \to_{\mathrm{select}} x$ in $\mathcal{L}\mathsf{ex}$, is a decomposition of the first rule into atomic $\mathcal{L}\mathsf{ex}$-steps. This also holds for the other rules of $\mathcal{D}\mathsf{ed}$. E.g. the case of the fourth rule is displayed in Figure 1.

Remark 1. The atomic decomposition TRS $\mathcal{L}\mathsf{ex}$ is not minimal; in general it does not yield unique atomic decompositions of rules. For instance, assuming for the moment that $\mathtt{M}\ R\ \mathtt{0}$ would hold, the third rule $\mathtt{M}(x, \mathtt{0}) \to \mathtt{0}$ of $\mathcal{D}\mathsf{ed}$

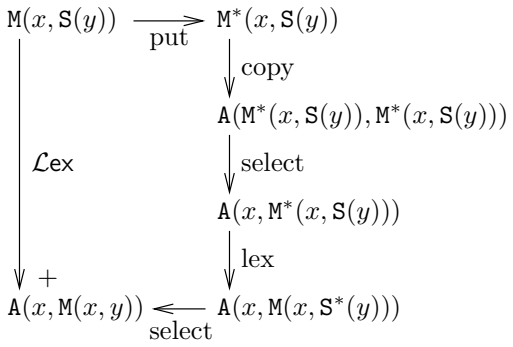

Fig. 1. Atomic $\mathcal{L}\mathsf{ex}$-decomposition of the fourth Dedekind rule

could be atomically decomposed into both $\mathtt{M}(x,\mathtt{0}) \to_{\mathrm{put}} \mathtt{M}^*(x,\mathtt{0}) \to_{\mathrm{select}} \mathtt{0}$ and $\mathtt{M}(x,\mathtt{0}) \to_{\mathrm{put}} \mathtt{M}^*(x,\mathtt{0}) \to_{\mathrm{copy}} \mathtt{0}$; the term $\mathtt{M}^*(x,\mathtt{0})$ is copied to all (zero!) arguments of the symbol $\mathtt{0}$.

Definition 2.

1. *The* iterative lexicographic path order R_{ilpo} *of a relation* R *on a signature* Σ *is the restriction of* $\to^+_{\mathcal{L}\mathsf{ex}}$ *to* $T(\Sigma \uplus V)$.
2. *A TRS is* ILPO-terminating *if its rules are contained in* R_{ilpo} *for some terminating relation* R.

For the TRS $\mathcal{D}\mathsf{ed}$ we already saw that $l \to^+_{\mathcal{L}\mathsf{ex}} r$ holds for each rule $l \to r$. Hence $\mathcal{D}\mathsf{ed}$ is ILPO-terminating.

An observation that plays a crucial role in many termination methods is that a TRS is terminating if and only if it admits a *reduction order*, i.e. iff its rules are contained in a terminating (order) relation which is closed under contexts and substitutions. (See e.g. [10, Prop. 6.1.3].) Note that transitivity and closure under contexts and substitutions of R_{ilpo} are 'built in' into its definition via the atomic decomposition TRS. Therefore, in order to show that R_{ilpo} is a reduction order, it only remains to be shown that it is terminating. This will be proved in Section 4. From that result we can then conclude that ILPO-termination implies termination.

But first, in Section 3, we present some further examples of ILPO-terminating TRSs.

Remark 2. Although by definition the iterative lexicographic path order R_{ilpo} is transitive, even in cases when R isn't, we do not put stress on this. In particular, transitivity is not used in the proof that termination lifts from R to R_{ilpo}.

Remark 3. The iterative lexicographic path order as presented here is a strengthening of the version of the iterative path order in [9] (which is there still called

recursive path order). The difference is that in [9] instead of the lex-rule (cf. Table 2) the *down*-rule is employed:

$$f^*(\boldsymbol{x}, g(\boldsymbol{y}), \boldsymbol{z}) \to_{\text{down}} f(\boldsymbol{x}, g^*(\boldsymbol{y}), \boldsymbol{z})$$

It expresses that a term may be made smaller by making one of its arguments smaller. The down-rule is a derived rule in our system:

$$f^*(\boldsymbol{x}, g(\boldsymbol{y}), \boldsymbol{z}) \to_{\text{lex}} f(\boldsymbol{x}, g^*(\boldsymbol{y}), \boldsymbol{l}) \twoheadrightarrow_{\text{select}} f(\boldsymbol{x}, g^*(\boldsymbol{y}), \boldsymbol{z})$$

where the ith select-step applies to the ith occurrence of $l = f^*(\boldsymbol{x}, g(\boldsymbol{y}), \boldsymbol{z})$, and then selects z_i. The implication in the other direction does not hold as witnessed by the one-rule TRS $f(a, b) \to f(b, a)$ which cannot be proven terminating by the method presented in [9], but which is ILPO-terminating for $a \; R \; b$:

$$f(a, b) \to_{\text{put}} f^*(a, b) \to_{\text{lex}} f(a^*, f^*(a, b)) \to_{\text{copy}} f(b, f^*(a, b)) \to_{\text{select}} f(b, a)$$

The *simplify-left-argument*-rule, introduced in the exercises in [9] in order to prove termination of the Ackermann TRS $\mathcal{A}\mathsf{ck}$ (see below), is also easily derived in our system: it simply is the lex-rule with $\boldsymbol{x}$ taken to be the empty vector, i.e. the leftmost argument must be made smaller. It is easy to see that also that version is strictly weaker than ILPO-termination.

3 Examples of ILPO-terminating TRSs

In this section the iterative lexicographic path order method is illustrated by applying it to some well-known TRSs.

The example of Dedekind's rules for addition and multiplication only employs trivial applications of the lex-rule, where $\boldsymbol{z}$ is empty. Proving ILPO-termination of the Ackermann function requires non-trivial applications of the lex-rule.

Example 1 (Ackermann's function). The TRS $\mathcal{A}\mathsf{ck}$ has a signature consisting of the nullary symbol $\mathtt{0}$, the unary symbol $\mathtt{S}$, and the binary symbol $\mathtt{Ack}$, with rules as in Table 3.

$$\begin{aligned} \mathtt{Ack}(\mathtt{0}, y) &\to \mathtt{S}(y) \\ \mathtt{Ack}(\mathtt{S}(x), \mathtt{0}) &\to \mathtt{Ack}(x, \mathtt{S}(\mathtt{0})) \\ \mathtt{Ack}(\mathtt{S}(x), \mathtt{S}(y)) &\to \mathtt{Ack}(x, \mathtt{Ack}(\mathtt{S}(x), y)) \end{aligned}$$

Table 3. Ackermann's function

For the relation R defined by $\mathtt{Ack} \; R \; \mathtt{S}$, the TRS $\mathcal{A}\mathsf{ck}$ is ILPO-terminating as witnessed by the following atomic decompositions of each of its rules:

- $\mathtt{Ack}(\mathtt{0}, y) \to_{\text{put}} \mathtt{Ack}^*(\mathtt{0}, y) \to_{\text{copy}} \mathtt{S}(\mathtt{Ack}^*(\mathtt{0}, y)) \to_{\text{select}} \mathtt{S}(y)$.
- $\mathtt{Ack}(\mathtt{S}(x), \mathtt{0}) \to_{\text{put}} \mathtt{Ack}^*(\mathtt{S}(x), \mathtt{0}) \to_{\text{lex}}$
  $\mathtt{Ack}(\mathtt{S}^*(x), \mathtt{Ack}^*(\mathtt{S}(x), \mathtt{0})) \to_{\text{select}} \mathtt{Ack}(x, \mathtt{Ack}^*(\mathtt{S}(x), \mathtt{0})) \to_{\text{copy}}$
  $\mathtt{Ack}(x, \mathtt{S}(\mathtt{Ack}^*(\mathtt{S}(x), \mathtt{0}))) \to_{\text{select}} \mathtt{Ack}(x, \mathtt{S}(\mathtt{0}))$.

– $\mathtt{Ack}(\mathtt{S}(x),\mathtt{S}(y)) \to_{\mathrm{put}} \mathtt{Ack}^*(\mathtt{S}(x),\mathtt{S}(y)) \to_{\mathrm{lex}}$
$\mathtt{Ack}(\mathtt{S}^*(x),\mathtt{Ack}^*(\mathtt{S}(x),\mathtt{S}(y))) \to_{\mathrm{select}} \mathtt{Ack}(x,\mathtt{Ack}^*(\mathtt{S}(x),\mathtt{S}(y))) \to_{\mathrm{lex}}$
$\mathtt{Ack}(x,\mathtt{Ack}(\mathtt{S}(x),\mathtt{S}^*(y))) \to_{\mathrm{select}} \mathtt{Ack}(x,\mathtt{Ack}(\mathtt{S}(x),y))$.

Example 2 (Dershowitz and Jouannaud [12]). Consider the string rewrite system $\mathcal{DJ}$ given by the four rules in Table 4.

$\mathtt{10} \to \mathtt{0001}$
$\mathtt{01} \to \mathtt{1}$
$\mathtt{11} \to \mathtt{0000}$
$\mathtt{00} \to \mathtt{0}$

Table 4. String rewrite system on 0,1-words

So we have e.g. the reduction

$\mathtt{1101} \to \mathtt{100011} \to \mathtt{10011} \to \mathtt{0001011} \to \mathtt{001011} \to \mathtt{00100000} \to \mathtt{0000010000} \to \ldots$

To capture this string rewrite system as a term rewrite system, the symbols 0 and 1 are perceived as unary function symbols and the rules are read accordingly; e.g. $\mathtt{10} \to \mathtt{0001}$ is the term rewrite rule

$$\mathtt{1}(\mathtt{0}(x)) \to \mathtt{0}(\mathtt{0}(\mathtt{0}(\mathtt{1}(x))))$$

To show ILPO-termination for $\mathcal{DJ}$, we set $\mathtt{1}\ R\ \mathtt{0}$ and check $l\ R_{ilpo}\ r$ for every rule $l \to r$. For the displayed rule the corresponding atomic decomposition is shown in Figure 2 (after dropping all parentheses).

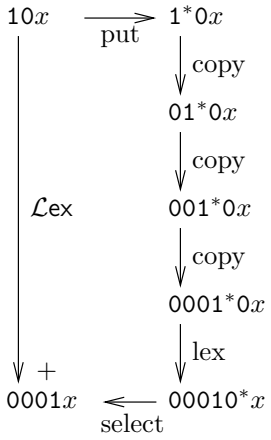

Fig. 2. Atomic $\mathcal{L}\mathsf{ex}$-decomposition of the rule $\mathtt{10}x \to \mathtt{0001}x$

Example 3 (Primitive recursion). Let a TRS on the natural numbers be given, having a unary function symbol g and a ternary function symbol h. Suppose we adjoin a binary symbol f to the signature, with defining rules

$$\begin{aligned} f(\mathtt{0}, x) &\to g(x) \\ f(\mathtt{S}(x), y) &\to h(f(x, y), x, y) \end{aligned}$$

If the original TRS is ILPO-terminating, say for the terminating relation R on its signature, then the resulting system is ILPO-terminating again as can be easily seen after adjoining $f\ R\ g$ and $f\ R\ h$ to R (this yields a terminating relation again).

4 ILPO-termination implies termination

The remaining task is now to show that if a relation R is terminating, then R_{ilpo} is terminating as well. As explained in Section 2, R_{ilpo} is then a reduction order, and it follows that ILPO-termination implies termination.

Since the rewrite rules of R_{ilpo} are given by the restriction of $\to^+_{\mathcal{L}\mathsf{ex}}$ to $T(\Sigma \uplus V)$, termination of the atomic decomposition TRS $\mathcal{L}\mathsf{ex}$ would be sufficient for termination of R_{ilpo}. However, $\mathcal{L}\mathsf{ex}$ is in general *not* terminating. For instance, in case of the running example we have, despite R being terminating:

$$\mathtt{A}(x, y) \to_{\text{put}} \mathtt{A}^*(x, y) \to_{\text{copy}} \mathtt{S}(\mathtt{A}^*(x, y)) \to_{\text{copy}} \mathtt{S}(\mathtt{S}(\mathtt{A}^*(x, y))) \to_{\text{copy}} \dots$$

We even have cycles

$$\mathtt{A}^*(x, y) \to_{\text{copy}} \mathtt{S}(\mathtt{A}^*(x, y)) \to_{\text{put}} \mathtt{S}^*(\mathtt{A}^*(x, y)) \to_{\text{select}} \mathtt{A}^*(x, y)$$

In either case, non-termination is 'caused' by the left-hand side of the copy-rule being a subterm of its right-hand side; *a priori* such an iteration is not bounded. Similar examples can be given with the lex-rule, which is also self-embedding.

However, observe that in both of the infinite reductions the control symbol $\mathtt{A}^*$ is 'used' infinitely often — by the copy rule. We will show that this is necessary in *any* infinite reduction. More precisely, that if for each control symbol a bound is given in advance on how often it can be used in the copy- and lex-rules, this will yield a terminating TRS $\mathcal{L}\mathsf{ex}^\omega$. Since in any given atomic decomposition $l \to^+_{\mathcal{L}\mathsf{ex}} r$ of a rule $l \to r$, any control symbol is only used finitely often (certainly not more often than the length of the decomposition), the relations $\to^+_{\mathcal{L}\mathsf{ex}}$ and $\to^+_{\mathcal{L}\mathsf{ex}^\omega}$ coincide on the unmarked terms. We will exploit this fact by proving termination of R_{ilpo} *via* termination of $\mathcal{L}\mathsf{ex}^\omega$.

Definition 3. *Let R be a relation on a signature Σ, and let V be a signature of nullary symbols disjoint from Σ. The TRS $\mathcal{L}\mathsf{ex}^\omega$ is $\langle \Sigma \uplus \Sigma^\omega \uplus V, \mathcal{R}^\omega \rangle$:*

1. *The signature Σ^ω of* ω-control *symbols consists of ω copies of Σ, i.e. for each symbol $f \in \Sigma$ and natural number n, Σ^ω contains a fresh symbol f^n having the arity f has.*

$$
\begin{array}{rll}
f(\boldsymbol{x}) & \to_{\mathrm{put}} & f^n(\boldsymbol{x}) \\
f^n(\boldsymbol{x}) & \to_{\mathrm{select}} & x_i \quad (1 \le i \le |\boldsymbol{x}|) \\
f^{n+1}(\boldsymbol{x}) & \to_{\mathrm{copy}} & g(f^n(\boldsymbol{x}), \ldots, f^n(\boldsymbol{x})) \quad (f\ R\ g) \\
f^{n+1}(\boldsymbol{x}, g(\boldsymbol{y}), \boldsymbol{z}) & \to_{\mathrm{lex}} & f(\boldsymbol{x}, g^n(\boldsymbol{y}), l, \ldots, l) \quad (l = f^n(\boldsymbol{x}, g(\boldsymbol{y}), \boldsymbol{z}))
\end{array}
$$

Table 5. The rules of $\mathcal{L}\mathsf{ex}^\omega$

2. *The rules $\mathcal{R}^\omega$ are given in the table, for arbitrary symbols f, g in Σ and natural number n, with $\boldsymbol{x}$, $\boldsymbol{y}$, $\boldsymbol{z}$ disjoint vectors of pairwise disjoint variables of appropriate lengths.*

The TRS $\mathcal{L}\mathsf{ex}$ (Definition 1) is seen to be a homomorphic image of $\mathcal{L}\mathsf{ex}^\omega$ by mapping f^n to f^*, for any natural number n. Vice versa, reductions in $\mathcal{L}\mathsf{ex}$ can be 'lifted' to $\mathcal{L}\mathsf{ex}^\omega$.

Lemma 1. *$\to^+_{\mathcal{L}\mathsf{ex}}$ and $\to^+_{\mathcal{L}\mathsf{ex}^\omega}$ coincide as relations restricted to $T(\Sigma \uplus V)$.*

Proof.

($\subseteq$) One shows that $\to^+_{\mathcal{L}\mathsf{ex}}$ is included in $\to^+_{\mathcal{L}\mathsf{ex}^\omega}$ by formalizing the reasoning already indicated above. In particular, one can translate (*lift*) a finite reduction of length n in $\mathcal{L}\mathsf{ex}$ to $\mathcal{L}\mathsf{ex}^\omega$ by replacing all marks ($*$) in the begin term by a natural number greater than n and, along the reduction, likewise each $*$ introduced by an application of the put-rule. Numerical values for the other marks then follow automatically by applying the $\mathcal{L}\mathsf{ex}^\omega$-rules that correspond to the original $\mathcal{L}\mathsf{ex}$-rules.
The result then follows, because, if $t \to^+_{\mathcal{L}\mathsf{ex}} s$ and the terms t, s are in $T(\Sigma \uplus V)$, i.e. do not contain marks, then the transformation of a $\to_{\mathcal{L}\mathsf{ex}}$-reduction from t to s to $\mathcal{L}\mathsf{ex}^\omega$ leaves the begin and end terms t and s untouched.

($\supseteq$) Every $\to^+_{\mathcal{L}\mathsf{ex}^\omega}$ step is a $\to^+_{\mathcal{L}\mathsf{ex}}$ step, by the homomorphism. □

Example 4. The atomic decomposition displayed in Figure 1 can be lifted to:

– $\mathtt{M}(x, \mathtt{S}(y)) \to_{\mathrm{put}} \mathtt{M}^2(x, \mathtt{S}(y)) \to_{\mathrm{copy}} \mathtt{A}(\mathtt{M}^1(x, \mathtt{S}(y)), \mathtt{M}^1(x, \mathtt{S}(y))) \to_{\mathrm{select}} \mathtt{A}(x, \mathtt{M}^1(x, \mathtt{S}(y))) \to_{\mathrm{lex}} \mathtt{A}(x, \mathtt{M}(x, \mathtt{S}^0(y))) \to_{\mathrm{select}} \mathtt{A}(x, \mathtt{M}(x, y))$.

The main theorem is proven by employing an ingenious (constructive) proof technique due to Buchholz [5].[6]

Lemma 2. *If R is terminating, then $\mathcal{L}\mathsf{ex}^\omega$ is terminating.*

Proof. To make the notation uniform, in the sense that all function symbols will carry a label, we employ $f^\omega(\boldsymbol{t})$ to denote the term $f(\boldsymbol{t})$ for an *unmarked* symbol

[6] The technique has been discovered independently by Jouannaud and Rubio [13], who show that it combines well with the Tait–Girard reducibility technique (both are essentially based on induction on terms), leading to a powerful termination proof technique also applicable to higher-order term rewriting.

f. This allows us to write *any* $\mathcal{L}\mathsf{ex}^\omega$-term uniquely as an *$\omega$-marked term* of the form $f^\alpha(\boldsymbol{t})$ for some unmarked symbol f, some ordinal α and some vector of terms $\boldsymbol{t}$. The ordinal α will be a natural number n or ω. In the crucial induction in this proof we will make use of the fact that in the ordering of the ordinals we have $\omega > n$ for each natural number n.

We prove by induction on the construction of terms that any ω-marked term is terminating. To that end it is sufficient to show that any ω-marked term $f^\alpha(\boldsymbol{t})$ is terminating under the assumption (the induction hypothesis) that its arguments $\boldsymbol{t}$ are terminating.

So assume that $t_1, \ldots, t_n$ are terminating, with n the arity of f. We prove that $f^\alpha(\boldsymbol{t})$ is terminating by a further induction on the triple consisting of f, $\boldsymbol{t}$, and α in the lexicographic product of the relations R, $(\to_{\mathcal{L}\mathsf{ex}^\omega} \restriction \mathcal{SN})^n$ and $>$. Here $(\to_{\mathcal{L}\mathsf{ex}^\omega} \restriction \mathcal{SN})^n$ is the n-fold lexicographic product of the terminating part of $\to_{\mathcal{L}\mathsf{ex}^\omega}$, with n the arity of f.

Clearly, the term $f^\alpha(\boldsymbol{t})$ is terminating if all its one-step $\to_{\mathcal{L}\mathsf{ex}^\omega}$-reducts are.[7] The latter we prove by distinguishing cases on the type of the reduction step.

1. If the step is a head step, we perform a further case analysis on the rule applied.
   (put) The result follows by the IH for the third component of the triple, since $\omega > m$ for any natural number m.
   (select) The result follows by the termination assumption for the $\boldsymbol{t}$.
   (copy) Then α is of the form $m + 1$ for some natural number m, and the reduct has shape $g^\omega(f^m(\boldsymbol{t}), \ldots, f^m(\boldsymbol{t}))$ for some g such that $f\ R\ g$. By the IH for the third component, each of the $f^m(\boldsymbol{t})$ is terminating. Hence, by the IH for the first component, the reduct is terminating.
   (lex) Then α is of the form $m+1$ for some natural number m and the reduct has shape $f^\omega(t_1, \ldots, t_{i-1}, g^m(\boldsymbol{s}), f^m(\boldsymbol{t}), \ldots, f^m(\boldsymbol{t}))$, with $t_i = g^\omega(\boldsymbol{s})$. Each $f^m(\boldsymbol{t})$ is terminating by the IH for the third component. Hence, the reduct is terminating by the IH for the second component, since $g^m(\boldsymbol{s})$ is a one-step $\mathcal{L}\mathsf{ex}^\omega$-reduct of t_i (for the put-rule).
2. If the step is a non-head step, then it rewrites some direct argument, and the result follows by the IH for the second component. □

Theorem 1. *ILPO-termination implies termination.*

Proof. Suppose the TRS $\mathcal{T}$ is ILPO-terminating for some terminating relation R on its signature, i.e. $l\ R_{ilpo}\ r$ holds for each rule $l \to r$ in $\mathcal{T}$.

Since R_{ilpo} is defined as a restriction of $\to^+_{\mathcal{L}\mathsf{ex}}$ which in turn coincides with $\to^+_{\mathcal{L}\mathsf{ex}^\omega}$ by Lemma 1, it is a transitive relation that is closed under contexts and substitutions and, by Lemma 2, also terminating. Hence, R_{ilpo} is a reduction order, and therefore $\mathcal{T}$ must be terminating. □

[7] This observation can be used as an inductive characterization of termination: a term is terminating if and only if all its one-step reducts are. In a constructive rendering of the proof one can take this characterization as the definition of termination.

Hence termination of the example TRSs such as $\mathcal{D}\mathsf{ed}$, follows from their ILPO-termination as established above.

Remark 4. It is worth noting that Buchholz's proof technique can also be applied to non-simplifying TRSs. For instance, for proving termination of the one-rule TRS $f(f(x))\to f(g(f(x)))$ the technique boils down to showing that any instance $f(g(f(t)))$ of the right-hand side is terminating on the assumption that the direct subterm $f(t)$ of the corresponding instance of the left-hand side is. This follows by 'induction' on the right-hand side and cases on the shape of the left-hand side of the rule: $f(g(f(t)))$ can be rewritten neither at the head nor at position 1, hence it is terminating if $f(t)$ is.

5 Equivalence of ILPO with the recursive lexicographic path order

We show that ILPO is at least as powerful as the recursively defined lexicographic path order found in the literature, and is equivalent to it for transitive relations. The following definition of $>_{lpo}$ for a given strict order $>$ on the signature Σ, is copied verbatim from Definition 5.4.12 in the textbook by Baader and Nipkow [7].

Definition 4. *Let Σ be a* finite signature *and $>$ be a strict order on Σ. The* ***lexicographic path order*** *$>_{lpo}$ on $T(\Sigma, V)$ induced by $>$ is defined as follows: $t >_{lpo} s$ iff*

- ***(LPO1)*** *$s \in \mathcal{V}ar(t)$ and $t \neq s$, or*
- ***(LPO2)*** *$t = f(t_1, \ldots, t_m)$, $s = g(s_1, \ldots, s_n)$, and*
  - ***(LPO2a)*** *there exists i, $1 \leq i \leq m$, with $t_i \geq_{lpo} s$, or*
  - ***(LPO2b)*** *$f > g$ and $t >_{lpo} s_j$ for all j, $1 \leq j \leq n$, or*
  - ***(LPO2c)*** *$f = g$, $t >_{lpo} s_j$ for j, $1 \leq j \leq n$, and there exists i, $1 \leq i \leq m$, such that $t_1 = s_1$, …, $t_{i-1} = s_{i-1}$ and $t_i >_{lpo} s_i$.*

It is easy to see that this is still a correct recursive definition for $>$ being an arbitrary relation R, yielding R_{lpo}. We call a TRS $\mathcal{T} = \langle \Sigma, \mathcal{R} \rangle$ *LPO*-terminating for a terminating relation R, if $\mathcal{R} \subseteq R_{lpo}$.

Lemma 3. *$R_{lpo} \subseteq R_{ilpo}$, for any relation R.*

Proof. We show by induction on the definition of $t\ R_{lpo}\ s$ that $t^* \twoheadrightarrow_{\mathcal{L}\mathsf{ex}} s$, where $(f(\boldsymbol{t}))^* = f^*(\boldsymbol{t})$. This suffices, since $t \to_{\mathrm{put}} t^*$ and t, s are not marked.

- **(LPO1)** If $s \in \mathcal{V}ar(t)$ and $t \neq s$, then the result follows by repeatedly selecting the subterm on a path to an occurrence of s in t.
- **(LPO2)** Otherwise, let $t = f(t_1, \ldots, t_m)$, $s = g(s_1, \ldots, s_n)$.
  - **(LPO2a)** Suppose there exists i, $1 \leq i \leq m$, with either $t_i = s$ or $t_i\ R_{lpo}\ s$. In the former case, the result follows by a single application of the select-rule for index i. In the latter case, this step is followed by an application of the put-rule after which the result follows by the IH.

(LPO2b) Suppose $f\ R\ g$ and $t\ R_{lpo}\ s_j$ for all j, $1 \leq j \leq n$. Then the result follows by a single application of the copy-rule and n applications of the IH.

(LPO2c) Suppose $f = g$, $t\ R_{lpo}\ s_j$ for j, $1 \leq j \leq n$, and there exists i, $1 \leq i \leq m$, such that $t_1 = s_1$, …, $t_{i-1} = s_{i-1}$ and $t_i\ R_{lpo}\ s_i$. Then the result follows by a single application of the lex-rule, selecting the ith argument, and the IH for $t_i\ R_{lpo}\ s_i$ and $t\ R_{lpo}\ s_j$ for j, $i < j \leq n$. □

We call the $\mathcal{L}$ex-strategy implicit in this proof the *wave* strategy. The idea is that the marked positions in a term represent the wave front, which moves downwards, i.e. from the root in the direction of the subtrees of a left-hand side, generating an ever growing prefix of the right-hand side behind it. This is visualised abstractly in Figure 3, and for the atomic $\mathcal{L}$ex-decomposition of $\mathtt{M}(x, \mathtt{S}(y)) \to \mathtt{A}(x, \mathtt{M}(x, y))$

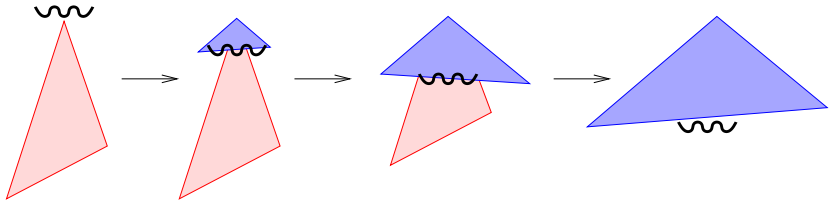

Fig. 3. Wave strategy

of Figure 1, in Figure 4. (In fact, all $\mathcal{L}$ex-reductions given above adhere to the wave strategy.) One can prove a converse to Lemma 3 by a detailed proof-

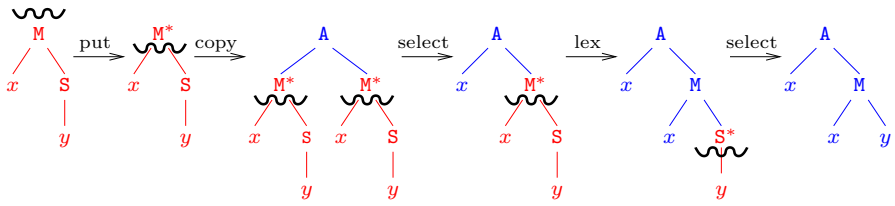

Fig. 4. Wave strategy for atomic $\mathcal{L}$ex-decomposition of $\mathtt{M}(x, \mathtt{S}(y)) \to \mathtt{A}(x, \mathtt{M}(x, y))$

theoretic analysis, showing that any $\mathcal{L}$ex-reduction can be transformed, into a wave reduction. The upshot is that in general $R_{ilpo} = (R_{lpo})^+$. As a corollary we then have that ILPO is equivalent with LPO for any strict order, and that

R_{ilpo} is decidable, in case R is a terminating relation for which reachability is decidable: simply 'try all waves up to the size of the right-hand side'.[8]

Remark 5. Note that $(R^+)_{ilpo}$ may differ from R_{ilpo}. Consider the signature consisting of nullary symbols a, b, and unary symbols f, g, and the terminating relation $f\ R\ b\ R\ g$. Then the one-rule TRS $f(a) \to g(a)$ is not ILPO-terminating, but it is ILPO-terminating for R^+. The problem is that making $f(a)$ smaller using R forces erasure of its argument a, because b is nullary.

A proof-theoretic analysis of the wave strategy is beyond the scope of this paper. Here we will be satisfied by giving a rather *ad hoc* proof of the converse of Lemma 3, for the case where we start with a transitive relation R.

Lemma 4. *$R_{ilpo} \subseteq R_{lpo}$, for any transitive relation R.*

Proof. Fix the relation R. By definition, if $t\ R_{ilpo}\ s$ then $t \to^+_{\mathcal{L}\mathsf{ex}} s$. By Lemma 1, then also $t \to^+_{\mathcal{L}\mathsf{ex}^\omega} s$. To show that this implies $t\ R_{lpo}\ s$, we employ a homomorphic embedding ϵ of ω-marked terms (as introduced in the proof of Lemma 2) defined by $f^\alpha(\boldsymbol{u}) \mapsto f(\epsilon(\boldsymbol{u}), \alpha)$. Here the terms in the range of ϵ are terms over the signature obtained from Σ by increasing the arity of every function symbol by 1, and adjoining the ordinals up to ω as nullary symbols. The idea of the embedding is that every function symbol gets an extra final argument signifying how many times the symbol may be 'used'. Initially (unmarked) it is set to ω. Embedding the TRS $\mathcal{L}\mathsf{ex}^\omega$ (see Table 5) yields the TRS $\epsilon(\mathcal{L}\mathsf{ex}^\omega)$ having rules given in Table 6.

$$
\begin{array}{rcl}
f(\boldsymbol{x}, \omega) & \to_{\mathrm{put}} & f(\boldsymbol{x}, n) \\
f(\boldsymbol{x}, n) & \to_{\mathrm{select}} & x_i \quad (1 \leq i \leq |\boldsymbol{x}|) \\
f(\boldsymbol{x}, n+1) & \to_{\mathrm{copy}} & g(f(\boldsymbol{x}, n), \ldots, f(\boldsymbol{x}, n), \omega) \quad (f\ R\ g) \\
f(\boldsymbol{x}, g(\boldsymbol{y}, \omega), \boldsymbol{z}, n+1) & \to_{\mathrm{lex}} & f(\boldsymbol{x}, g(\boldsymbol{y}, n), \boldsymbol{l}, \omega) \quad (l = f(\boldsymbol{x}, g(\boldsymbol{y}, \omega), \boldsymbol{z}, n))
\end{array}
$$

Table 6. Rules of $\epsilon(\mathcal{L}\mathsf{ex}^\omega)$

By definition of ϵ, $t \to^+_{\mathcal{L}\mathsf{ex}^\omega} s$ implies $\epsilon(t) \to^+_{\epsilon(\mathcal{L}\mathsf{ex}^\omega)} \epsilon(s)$. It is easy to verify that each of the $\epsilon(\mathcal{L}\mathsf{ex}^\omega)$-rules is contained in $\epsilon(R)_{lpo}$, where $\epsilon(R)$ is obtained by taking the union of R and the natural greater than relation $>$ on the ordinal symbols, and relating every function symbol to any ordinal symbol. Note that $\epsilon(R)$ is transitive since R and $>$ are. Since the relation $\epsilon(R)_{lpo}$ is closed under contexts and substitutions and is transitive if $\epsilon(R)$ is (see e.g. [10, 7]),[9] we conclude that

[8] This is somehow analogous to the way in which *rippling* guides the search for a proof of a goal from a given [14].

[9] Note that these properties need to be verified separately for the *recursive* lexicographic path order (or any variation on it). In case of the *iterative* lexicographic path order (or any variation on it), these properties hold automatically by it being given by a TRS, allowing one to focus on establishing termination.

$\epsilon(t)\ \epsilon(R)_{lpo}\ \epsilon(s)$. That this implies $t\ R_{lpo}\ s$, follows by an easy induction on the definition of the former. The crux of the proof is that the adjoined ordinal symbols do not relate to the original symbols from Σ. The only problematic cases (since then the IH could not be applied) are:

(LPO2a) holds since either $\omega = \epsilon(s)$ or $\omega\ \epsilon(R)_{lpo}\ \epsilon(s)$. Neither can be the case since ω is not related to symbols in Σ, in particular not to the head symbol of $\epsilon(s)$ (which is the same as the head of s).

(LPO2c) holds since $\omega\ \epsilon(R)_{lpo}\ \omega$. This obviously cannot be the case.

In the other cases the IH does the job, e.g.

(LPO2a) holds since either $\epsilon(t_i) = \epsilon(s)$, or $\epsilon(t_i)\ \epsilon(R)_{lpo}\ \epsilon(s)$, for some i. Then either $t_i = s$ by injectivity of ϵ, or $t_i\ \epsilon(R)_{lpo}\ s$ by the IH, and we conlude $t\ R_{lpo}\ s$. □

Combining Lemmas 3 and 4 yields our second main result.

Theorem 2. *$>_{ilpo} = >_{lpo}$, for any transitive relation (order) $>$.*

6 Conclusion

We have shown that our iterative set-up of ILPO can serve as an independent alternative to the classical recursive treatment of LPO. It can be seen as being obtained by *decomposing* the recursive definition, extracting *atomic* rules from the inductive clauses. From this perspective it is only natural that we have taken an arbitrary terminating relation (instead of order) on the signature as our starting point, so one could speak, in the spirit of Persson's presentation of recursive path relations [15], of iterative lexicographic path *relations*.

We claim that the correspondence between recursive and iterative ways of specifying path orders is robust, i.e. goes through for variants of LPO like the embedding relation and recursive path orders. Substantiating the claim is left to future research.

Another direction for further investigation is suggested by Remark 4. It seems that an analogous argument can be used to yield soundness of Arts and Giesl's *dependency-pair* technique for proving termination. (See e.g. [10, Section 6.5.5].) Thus, whereas non-simplifying TRSs are traditionally out of the scope of the recursive path order method, by their termination proof being tied to Kruskal's Tree Theorem, Buchholz's technique will give us a handle on a uniform treatment of both path orders and the dependency-pair technique.

References

1. Dershowitz, N.: Orderings for term rewriting systems. TCS **17**(3) (1982) 279–301
2. Marcone, A.: Fine analysis of the quasi-orderings on the power set. Order **18** (2001) 339–347

3. Bergstra, J., Klop, J.: Algebra of communicating processes. TCS **37**(1) (1985) 171–199
4. Bergstra, J., Klop, J., Middeldorp, A.: Termherschrijfsystemen. Programmatuurkunde. Kluwer (1989)
5. Buchholz, W.: Proof-theoretic analysis of termination proofs. APAL **75**(1-2) (1995) 57–65
6. Kamin, S., Lévy, J.J.: Two generalizations of the recursive path ordering. University of Illinois (1980)
7. Baader, F., Nipkow, T.: Term Rewriting and All That. Cambridge University Press (1998)
8. Geser, A.: Relative Termination. PhD thesis, Universität Passau, Germany (1990)
9. Klop, J.: Term rewriting systems. In Abramsky, S., Gabbay, D., Maibaum, T., eds.: Handbook of Logic in Computer Science. Volume 2, Background: Computational Structures. Oxford University Press (1992) 1–116
10. Terese: Term Rewriting Systems. Volume 55 of Cambridge Tracts in Theoretical Computer Science. Cambridge University Press (2003)
11. Dedekind, R.: Was sind und was sollen die Zahlen?, Brunswick (1888)
12. Dershowitz, N., Jouannaud, J.P.: Rewrite systems. In van Leeuwen, J., ed.: Handbook of Theoretical Computer Science. Volume B, Formal Models and Semantics. Elsevier (1990) 243–320
13. Jouannaud, J.P., Rubio, A.: The higher-order recursive path ordering. In: 14th Annual IEEE Symposium on Logic in Computer Science, IEEE Computer Society (1999) 402–411
14. Bundy, A., Basin, D., Hutter, D., Ireland, A.: Rippling: Meta-Level Guidance for Mathematical Reasoning. Volume 56 of Cambridge Tracts in Theoretical Computer Science. Cambridge University Press (2005)
15. Persson, H.: Type Theory and the Integrated Logic of Programs. PhD thesis, Chalmers, Sweden (1999)