LIF

Laboratoire d'Informatique Fondamentale
de Marseille

Unité Mixte de Recherche 6166
CNRS – Université de Provence – Université de la Méditerranée

A Constructive Axiomatization of the Recursive Path Ordering

Solange Coupet-Grimal and William Delobel

Rapport/Report 28-2006

9 January 2006

A Constructive Axiomatization of the Recursive Path Ordering

Solange Coupet-Grimal and William Delobel

LIF – Laboratoire d'Informatique Fondamentale de Marseille
UMR 6166
CNRS – Université de Provence – Université de la Méditerranée

CMI, 39 rue Joliot-Curie, F-13453, Marseille, France.

{Solange.Coupet, Delobel}@cmi.univ-mrs.fr.

Abstract/Résumé

We give an axiomatization of Recursive Path Orders in the Calculus of Inductive Constructions. Then, we show that they are monotonic strict partial orders, and that they are well-founded. The proof of the well-foundedness is particularly short and elementary. Finally, we produce three relations that are proved to model the axiomatization: the Multiset Path Ordering, the Lexicographic Path Ordering, and the Recursive Path Ordering with *status*. All this work is implemented in the Coq proof assistant.

Keywords: Term Rewriting Systems, Termination, Constructive Proofs, Theorem Proving, Coq.

Nous présentons une axiomatisation constructive des Ordres Récursifs sur les Chemins. Puis, nous démontrons que ce sont des ordres stricts, monotones et bien-fondés. La preuve de bonne fondation est particulièrement courte et élémentaire. Enfin, nous produisons trois modèles: l'ordre multi-ensemble sur les chemins, l'ordre lexicographique sur les chemins, et l'ordre récursif sur les chemins avec *status*. Tout ce travail est implémenté dans l'assistant de preuves Coq.

Mots-clés : Systèmes de réécriture sur les termes, Terminaison, Preuves constructives, Assistant de preuves, Coq.

Relecteurs/Reviewers: Frédéric Blanqui and Pierre Lescanne.

1 Introduction

This paper presents a constructive axiomatization in the Calculus of Inductive Constructions (CIC) of Recursive Path Orders (RPO). These orders compare first order terms by first comparing their root, and then the lists of their immediate sub-terms. These lists can be compared either as multisets in the case of the Multiset Path Order (MPO), defined in the seminal work of Dershowitz [Der82], or lexicographically in the case of Lexicographic Path Order (LPO) due to Kamin and Levy [KL80], or by a relation depending on the *status* $\tau(f)$ of the root f (RPO with status, first considered by Lescanne [Les83]). RPO are powerful tools for proving the termination of term rewriting systems, due to their key property: the well-foundedness. In general, in literature, the proof of their well-foundedness relies on the fact that they are simplification orders, and therefore they contain the homeomorphic embedding that is a well partial order (and thus well-founded) from Kruskal's theorem [Kru60]. However, the proof of Kruskal's theorem is not constructive and only applies when the signature is finite.

This work falls into two parts. Firstly, we define axiomatically RPO with status and we study two kinds of properties that are clearly separated: the fact of being monotonic strict orders and the well-foundedness. In both cases we introduce conditions on the relations defined by the status $\tau(f)$ for the properties to hold. Secondly, we show that MPO, LPO, and RPO with lexicographic and multiset status are models of this axiomatization, by establishing that both status satisfy the required conditions.

The main points of our contribution we would like to emphasize are the following:
– The proof of the well-foundedness is elementary and quite short. It is simply a sequence of nested inductions.
– Both properties are perfectly independent. In particular, the well-foundedness does not require the transitivity.
– Our approach is general. It applies to any RPO for which the sufficient conditions for one or other or both of the properties are fulfilled. Moreover, the terms we consider are not supposed to be ground nor the signature to be finite.
– This entire work has been carried out in the Coq proof-assistant [Tea04] to be part of *CoLoR*, the *Coq Library on Rewriting and Termination (http://color.loria.fr)*. Our development is strongly structured thanks to the Coq module mechanism.

The paper is organized as follows. Section 2 is devoted to specifying in the CIC the notions of accessibility, well-foundedness, and well-founded induction. Section 3 deals with first order terms and the standard definition of RPO. In section 4, RPO are axiomatized and we prove that they are monotonic strict orders and that they are well-founded. In section 5, we show that RPO with the lexicographic and multiset status is a model of the axiomatization in section 4. We present related work and we conclude in section 6. The appendix describes the architecture of the Coq Library.

2 Well-Foundedness

Let $(A, <)$ be a set equipped with a binary relation. The key notion for expressing the well-foundedness of $<$ is *accessibility*. Intuitively, an element a of A is accessible for the relation $<$, and this is denoted by $(acc_< \ x)$, if and only if all descending chains starting with x are finite. In the CIC, this is expressed by the inductive definition (1), associated with the induction principle (2) below:

$$\frac{\forall y : A,\ y < x \ \rightarrow \ (acc_< \ y)}{(acc_< \ x)} \tag{1}$$

$$\frac{(acc_< \ x) \quad (\forall y : A,\ y < x \ \rightarrow \ (P \ y)) \ \rightarrow \ (P \ x)}{(P \ x)} \tag{2}$$

Moreover, a relation $<$ is *well-founded* if and only if all elements are accessible. So :

$$(WF \ <) : \ Prop := (\forall a : A),\ (acc_< \ a) \tag{3}$$

We also define the notion for an element to be accessible for the restriction of a relation $<$ to a subset of A, characterized by a predicate S.

$$\frac{\forall y : A,\ (S \ y) \ \rightarrow \ y < x \ \rightarrow \ (acc_{<_{|S}} \ y)}{(acc_{<_{|S}} \ x)} \tag{4}$$

Here is the related induction principle :

$$\frac{(acc_{<_{|S}} \ x) \quad (\forall y : A,\ (S \ y) \ \rightarrow \ y < x \ \rightarrow \ (P \ y)) \ \rightarrow \ (P \ x)}{(P \ x)} \tag{5}$$

Lastly, we define the notion of restricted well-foundedness by:

$$(WF_S \ <) : \ Prop := (\forall a : A),\ (S \ a) \rightarrow (acc_{<_{|S}} \ a) \tag{6}$$

3 Recursive Path Ordering (RPO)

RPO are binary relations on first order terms and we start by specifying this notion in the CIC.

3.1 First order terms

Let us consider a set F of functional symbols, equipped with a preorder (i.e. a reflexive and transitive relation) $\leq_F$. As a preorder, $\leq_F$ contains an equivalence relation :

$$=_F := \ \lambda f, g.\ f \leq_F g \wedge g \leq_F f$$

and a strict partial order :

$$<_F := \ \lambda f, g.\ f \leq_F g \wedge f \neq_F g \quad .$$

We define inductively the type *term* of first order terms on the signature F and the set X of variable names, by the rules :

$$\frac{x : X}{(Var \ x) : term} \qquad \frac{f : F \qquad ss : (list \ term)}{(App \ f \ ss) \ : \ term}$$

where the type $list$ is parameterized by a set A and classically defined by:

$$\frac{}{nil : (list\ A)} \qquad \frac{h : A \qquad ss : (list\ A)}{(cons\ h\ ss)\ :\ (list\ A)}$$

In the sequel, we shall use the usual simplified notations:

- $(s_1, \ldots, s_n)$ for $(cons\ s_1 (\ldots (cons\ s_n\ nil) \ldots))$
- $h :: ss$ for a list with head h and tail ss.
- $f(s_1, \ldots, s_n)$ for $(App\ f\ (s_1, \ldots, s_n))$
- x for $(Var\ x)$
- $Vars(s)$ for the set of the variables that occur in term s.
- $s \in ss$ to express that s is an element of list ss.

An induction principle associated with type $term$ can be stated by the following rule, in which P is a predicate on $term$.

$$\frac{\begin{array}{c}\forall x\ :\ X,\ P\,(Var\ x) \\ \forall f : F, \forall ss : (list\ term),\ (\forall s : term,\ s \in ss\ \rightarrow\ (P\ s))\ \rightarrow\ (P\ f(ss))\end{array}}{\forall s\ :\ term,\ (P\ s)} \qquad (7)$$

This principle is proved by induction on the size of the terms.

3.2 Standard Definition of RPO

The relation $<_{RPO}$ as described below compares two terms by first comparing their roots, and then the lists of their immediate subterms. The comparison relation on these lists depends on the root symbol. We thus assume the existence of a $status$ function $\tau\ :\ F\ \rightarrow\ (relation\ term)\ \rightarrow\ (relation\ (list\ term))$ compatible with the relation $=_F$. In the sequel $<^{\tau(f)}$ will denote $(\tau\ f\ <)$.

The Recursive Path Ordering $<_{RPO}$ is a relation on the terms defined recursively by the 3 rules below:

$$\frac{g <_F f \quad \forall i\ \in \{1, \ldots, m\},\ t_i <_{RPO}\ s}{g(t_1, \ldots, t_m)\ <_{RPO}\ s = f(s_1, \ldots, s_n)} \qquad (RPO_1)$$

$$\frac{f =_F g \quad (t_1, \ldots, t_m)\ <^{\tau(f)}_{RPO}\ (s_1, \ldots, s_n) \quad \forall i\ \in \{1, \ldots, m\},\ t_i <_{RPO}\ s}{g(t_1, \ldots, t_m) <_{RPO}\ s = f(s_1, \ldots, s_n)} \qquad (RPO_2)$$

$$\frac{\exists i\ \in \{1, \ldots, n\},\ t\ \leq_{RPO}\ s_i}{t\ <_{RPO}\ f(s_1, \ldots, s_n)} \qquad (RPO_3)$$

Classically, $\leq_{RPO}$ denotes the reflexive closure of $<_{RPO}$.

We cannot specify directly $<_{RPO}$ in the CIC by an inductive definition relying on the rules RPO_i, $i \in \{1, 2, 3\}$, since certain syntactical criteria for the definition to be accepted are not satisfied. Instead, we take an axiomatic approach in which the relation is just a parameter, assumed to be a fixpoint of the operator on the binary relations on $term$, say $\mathcal{F}$, related to the recursive standard definition above.

4 Axiomatization of RPO

The fact that a relation $<_{RPO}$ is a fixpoint of the operator $\mathcal{F}$ is expressed by four axioms as follows.

4.1 An Axiomatic Definition

The three introduction rules RPO_1, RPO_2, RPO_3 are considered as axioms that state that $\mathcal{F}(<_{RPO}) \subset <_{RPO}$. We introduce a fourth one, called RPO_{inv}, that expresses that $s <_{RPO} t$ can only be obtained from one of the three rules, in other words, that $<_{RPO} \subset \mathcal{F}(<_{RPO})$. So, all throughout this section, $<_{RPO}$ is a parameter, and simply denotes a binary relation on the first order terms that satisfies these four axioms.

4.2 Symbols Behavior in RPO

These axioms are sufficient to establish some preliminary facts, that are related to the behavior of functional symbols and variables with respect to $<_{RPO}$, and will be instrumental for the sequel.

Lemma 1 *Let s be a term, ss a list of terms, and f and g two functional symbols such that $f =_F g$. Then $s <_{RPO} f(ss) \rightarrow s <_{RPO} g(ss)$.*

Proof By nested inductions on terms $f(ss)$ and $g(ss)$, following principle (7).

Lemma 2 *Variables are minimal terms for the relation $<_{RPO}$.*

Proof This is immediate since no axiom among RPO_i, $i \in \{1, 2, 3\}$, makes it possible to derive $s <_{RPO} x$, where s is a term and x is a variable. □

Lemma 3 *For all terms s and all variables x, if $x <_{RPO} s$ then $x \in Vars(s)$.*

Proof By induction on s. □

Lemma 4 *For all terms s and all variables x, if $x \in Vars(s)$ and $x \neq s$, then $x <_{RPO} s$.*

Proof By induction on s. □

Lemma 5 *Let s and t be two terms. If $t \leq_{RPO} s$ then $Vars(t) \subset Vars(s)$.*

Proof By nested inductions on terms t and s.

4.3 A Monotonic Strict Partial Order

We now introduce the additional conditions on the *status* function τ for $<_{RPO}$ to be a monotonic strict order. We treat simultaneously the transitivity and the irreflexivity, which may seem surprising. This comes from the fact that these properties are not independent when $\tau(f)$ is the multiset order.

Theorem 6 *Let us assume that for all relations $<$ on the set of terms, for all functional symbols f, and for all lists of terms ss :*
$\forall s : term,\ s \in ss \rightarrow \neg(s < s) \wedge (\forall s_1, s_2 : term,\ s < s_1 \rightarrow s_1 < s_2 \rightarrow s < s_2)$
implies
$\neg(ss <^{\tau(f)} ss) \wedge (\forall ss_1, ss_2,\ ss <^{\tau(f)} ss_1 \rightarrow ss_1 <^{\tau(f)} ss_2 \rightarrow ss <^{\tau(f)} ss_2)$.
Then $<_{RPO}$ is irreflexive and transitive.

Proof We proceed by induction on the structure of s following principle (7). Assuming that the property is fulfilled by all the terms of a list ss, we must prove that $f(ss)$ also satisfies it. The proposition $\neg(f(ss) <_{RPO} f(ss))$ follows from the fact that $\neg(ss <_{RPO}^{\tau(f)} ss)$ by the induction hypothesis. For the second part of the goal:

$$\forall t, u : term,\ f(ss) <_{RPO} t \ \rightarrow t <_{RPO} u \ \rightarrow\ f(ss) <_{RPO} u$$

we do two nested inductions on the structure of terms t and u. The proof is quite long, and for lack of space we do not give it here. A detailed version can be found in [CD05], where it is performed in the particular case of the Multiset Path Order, but is quite similar to this one. Let us mention that one uses the preliminary results given in section 4.2. □

In the theorem below :
- $ss[p]$ stands for the element of index p in list ss. It is of type $(option\ term)$ since it can be undefined in case p is greater than the length of ss.
- $ss[p := t]$ is the list obtained from ss by replacing the element of index p by t.

Theorem 7 *Let us assume that for all relations $<$ on the set of the terms, for all functional symbols f, and for all lists of terms ss :*
$\forall s : term, \forall p : nat,\ ss[p] = (some\ s) \rightarrow \forall t : term,\ s < t \rightarrow\ ss <^{\tau(f)} ss[p := t]$.
Then $<_{RPO}$ is monotonic.

Proof By induction on the structure of s. □

4.4 Well-Foundedness of RPO

This section is devoted to the well-foundedness of $<_{RPO}$. Let us point out that it is independent from the previous one. As a matter of fact, the well-foundedness does not require any of the assumptions introduced in section 4.3.

Following Ferreira and Zantema's terminology ([FZ95]), we define the notion of $lifting$.

Definition 8 *Let A be a set with a binary relation $<$. A binary relation $<^{\lambda}$ on $(list\ A)$ is called a $lifting$ if and only if, for every well-founded part S of $(A, <)$, the restriction of $<^{\lambda}$ to the lists whose elements are in S is well-founded.*

Using the notion of restricted well-foundedness introduced in section 2 (definition (6)), this is simply expressed in the CIC by:

$$lifting := WF_{accs} \tag{8}$$

where predicate $accs$ is defined by: $accs := \lambda ss : (list\ A).\forall s \in ss, (acc_{<}\ s)$.

The $lifting$ definition is in fact parameterized by the set A and the relation $<$. As far as the well-foundedness of $<_{RPO}$ is concerned, we specialize the definition by taking $(A,\ <) = (term,\ <_{RPO})$. Therefore, predicate $accs$ is now:

$$accs := \lambda ss : (list\ term).\forall s \in ss, (acc_{<_{RPO}}\ s).$$

We can state now our main theorem.

Theorem 9 *Let us assume that :*
(i) the strict partial order $<_F$ on the functional symbols is well-founded

(ii) for all functional symbols f, $<_{RPO}^{\tau(f)}$ is a lifting.
Then, $<_{RPO}$ is well-founded.

Proof Let s be a term and let us prove that s is accessible for $<_{RPO}$. We proceed by induction on s (principle (7)).

- **Base Case** If s is a variable, from lemma 2 s is a minimal element, and thus it is accessible.
- **Induction Step** Let f be a functional symbol and ss be a list of terms. We have to prove $(acc_{<_{RPO}}\ f(ss))$ under the induction hypothesis:
  HInd$_1$: $(accs\ ss)$
  This amounts to proving that $\forall f : F,\ (P\ f)$ where predicate P is defined by:
  $P\ :=\ \lambda f : F. \forall ss :\ (list\ term),\ (accs\ ss)\ \rightarrow (acc_{<_{RPO}}\ f(ss))$.
  Let f be a functional symbol. From (i), f is accessible for this relation and then, using induction principle (2), we are led to prove $(P\ f)$ under the induction hypothesis
  Hind$_2$: $\forall g : F,\ g <_F f\ \rightarrow\ (P\ g)$.
  The goal can be written $\forall ss :\ (list\ term),\ (Q\ ss)$ where predicate Q is defined by $Q\ :=\ \lambda ss : (list\ term).\ (accs\ ss)\ \rightarrow (acc_{<_{RPO}}\ f(ss))$.
  Let ss be a list of terms such that $(accs\ ss)$. From hypothesis (ii), we know that ss is accessible for the relation $<_{RPO}^{\tau(f)}$ restricted to the subset defined by predicate $accs$. Consequently, using induction principle (5), we have to prove $(acc_{<_{RPO}}\ f(ss))$ under the induction hypothesis:
  HInd$_3$: $\forall ts :\ (list\ term),\ ts <_{RPO}^{\tau(f)}\ ss\ \rightarrow\ (Q\ ts)$.

  By definition, the goal is equivalent to $\forall t,\ (R\ t)$ where R is defined by:
  $R\ :=\ \lambda t : term. t\ <_{RPO}\ f(ss)\ \rightarrow\ (acc_{<_{RPO}}\ t)$
  Let us do an induction on term t.
  - **Base Case** If t is a variable, from lemma 2, t is a minimal element, and thus it is accessible.
  - **Induction Step** Let g be a functional symbol and ts a list of terms. Let us prove $(R\ g(ts))$ under the induction hypothesis:
    HInd$_4$: $\forall t : term,\ t \in ts\ \rightarrow\ (R\ t)$.
    By the definition of R, we have to establish that $(acc_{<_{RPO}}\ g(ts))$ under the hypothesis $g(ts)\ <_{RPO}\ f(ss)$. As $<_{RPO}$ satisfies the axiom RPO_{inv}, this assumption leads us to consider three cases, following the rule (RPO$_i$) from which the inequality is derived.

    Case RPO_3. In this case, $g(ts)$ is less or equal than an element s of ss. Since by **HInd**$_1$, s is accessible, so is $g(ts)$.

    In both other cases, we know that $\forall t : term,\ t \in ts\ \rightarrow\ t <_{RPO} f(ss)$, and therefore from hypothesis **HInd**$_4$, we can deduce that all t in ts are accessible for $<_{RPO}$, so we add to the context the hypothesis **H** : $(accs\ ts)$.

    Case RPO_1. As $g\ <_F\ f$, by using hypotheses **Hind**$_2$ and **H**, we are done.

Case RPO_2. The hypotheses are: $(*)\ f =_F g$
$(**)\ ts <^{\tau(f)}_{RPO} ss$

From $(*)$ and lemma 1, the goal is now $(acc_{<_{RPO}}\ f(ts))$. From $(**)$ and $\mathbf{HInd}_3$ we deduce that ts satisfies Q. Therefore, we can conclude by applying hypothesis $\mathbf{H}$. □

5 IRPO: A Model of RPO

We now define a model, denoted by $<_{IRPO}$, of the axiomatization presented in section 4. This not only ensures its consistency, but it also provides all the results we aim at obtaining in the particular case of the Recursive Path Order with lexicographic and multiset status. The cases of MPO and LPO are just simplications of this model and are treated similarly.

5.1 Inductive Definition of IRPO with Status

Let $(A, \sim)$ be a setoïd with a decidable equality.

Lexicographic Order We assume that $<$ is a binary relation on A, compatible with $\sim$. It induces a lexicographic order $<^{lex}$ on $(list\ A)$, inductively defined by:

$$\frac{s \sim s' \quad l <^{lex} l'}{s :: l <^{lex} s' :: l'} \qquad \frac{s < s' \quad (length\ l) = (length\ l')}{s :: l <^{lex} s' :: l'} \tag{9}$$

Multiset Order We give here a brief description of the Coq specification of the multiset order. For a more detailed presentation, we refer the interested reader to [CD05].
A standard way to define multisets is to consider them as total functions from A to the set $\mathbb{N}$ of the natural numbers, compatible with $\sim$. Given such a multiset M, for all elements a of A, $M(a)$ is called the *multiplicity* of a in M. By definition, an element a belongs to M if and only if its multiplicity in M is greater than 0. We will be handling here multisets whose elements are the (finitely many) immediate subterms of certain terms. Therefore, all the multisets we consider in this paper are finite, and this precision will be omitted in the sequel. As far as their Coq implementation is concerned, we refer to Koprowski's work [Kop04]. The author first gives an axiomatic specification, then he shows that it can be modeled by the set of the finite lists of elements of A. The axiomatization involves a parameter $Multiset$ for the type of the multisets, a multiplicity $mult : Multiset \to A \to nat$, and an equivalence relation $\sim_{mul}$. Given M and N of type $Multiset$, $M \sim_{mul} N$ holds if and only if for all element a of A $(mult\ M\ a) = (mult\ N\ a)$. Let us point out that, in absence of the axiom of extensionality, this does not implies the equality of $(mult\ M)$ and $(mult\ N)$. The axiomatization also includes the union and the difference operations and a special element $\emptyset$ for the empty multiset. The finiteness of the multisets is expressed by means of the following reasoning principle:

$$\frac{(P\ \emptyset) \qquad (\forall M : (Multiset\ A))(\forall a : A)\ (P\ M) \to (P\ M \cup \{\!\{a\}\!\})}{\forall M : (Multiset\ A)\ (P\ M)} \tag{10}$$

where $\{\!\{a\}\!\}$ is the multiset whose only element a has multiplicity 1. From these axiomatic definitions, several other operations are introduced. In particular, a function *list2multiset* transforms recursively each list into a multiset. We have added a function *multiset2list* which builds a list from a multiset M by induction on M (principle (10)), and we have proved that $(list2multiset\,(multiset2list\,M)) \sim_{mul} M$.

We assume now that $>$ is a binary relation on A, compatible with $\sim$. It induces a relation $\gg$ on the multisets of A. A multiset N is less than a multiset M if it is obtained by replacing some elements of M by smaller elements. This relation is precisely defined by induction as follows:

$$\frac{M \sim_{mul} Z \cup X \quad N \sim_{mul} Z \cup Y \quad \neg(X \sim_{mul} \emptyset) \quad (\forall y : A,\ y \in Y \ \rightarrow\ \exists x : A,\ x \in X \wedge x > y)}{M \gg N}$$

Let $\ll$ denote the transposed relation and let us define its inverse image by *list2multiset*. We obtain the following binary relation on the lists:

$$<^{mul} := \lambda l,\, l' : (list\,A).\, (list2multiset\,l) \ll (list2multiset\,l')$$

IRPO with Status The modeling of RPO is parameterized by the signature of the first order terms, that is $S = (X, (F, \leq_F))$ as described in section 3.1. We introduce the set $name = \{lexicographic, multiset\}$ of the status names, and a parameter $status : F \rightarrow name$ which is assumed to be compatible with $=_F$. The relation $<_{IRPO}$ is then inductively defined in Coq by the four introduction rules below:

$$\frac{g <_F f \quad \forall i \in \{1,\ldots,m\},\, t_i <_{IRPO} f(s_1,\ldots,s_n)}{g(t_1,\ldots,t_m) <_{IRPO} f(s_1,\ldots,s_n)} \qquad (IRPO_1)$$

$$\frac{f =_F g \quad \forall i \in \{1,\ldots,m\},\, t_i <_{IRPO} f(s_1,\ldots,s_n) \quad (status\ f) = lexicographic \quad (t_1,\ldots,t_m) <^{lex}_{IRPO} (s_1,\ldots,s_n)}{g(t_1,\ldots,t_m) <_{IRPO} f(s_1,\ldots,s_n)} \qquad (IRPO_{2,lex})$$

$$\frac{f =_F g \quad (status\ f) = multiset \quad (t_1,\ldots,t_m) <^{mul}_{IRPO} (s_1,\ldots,s_n)}{g(t_1,\ldots,t_m) <_{IRPO} f(s_1,\ldots,s_n)} \qquad (IRPO_{2,mul})$$

$$\frac{\exists i \in \{1,\ldots,n\},\, t \leq_{IRPO} s_i}{t <_{IRPO} f(s_1,\ldots,s_n)} \qquad (IRPO_3)$$

From this inductive definition we can prove straightforwardly that $<_{IRPO}$ fulfills the four axioms of section 4.1.

5.2 IRPO is a Monotonic Strict Order

It is sufficient to verify that the relations $<^{lex}$ and $<^{mul}$ defined in section 5.1 satisfy the hypotheses of theorems 6 and 7. For the lexicographic order, the proofs are standard. Let us just mention that the two first are performed by induction on the list on which the statements are universally quantified. For the multiset order, the proofs are also simple and do not deserve any comment (see [CD05] for more details).

5.3 IRPO is Well-Founded

Assuming that the precedence on the functional symbols is well-founded, we must prove that both the lexicographic order and the multiset order induced by $<_{IRPO}$ are $liftings$. This follows from the two next lemmas.

Lemma 10 *For all sets A and for all binary relations $<$ on A, $<^{lex}$ is a lifting.*

Preliminaries. Let us define $accs := \lambda l : (list\ A).\forall a \in l, (acc_< a)$. We have to prove that the relation $<^{lex}$ has the following property:

$$\forall l : (list\ A),\ (accs\ l)\ \rightarrow\ (acc_{<^{lex}_{|accs}}\ l)\quad .$$

In fact we change the goal into:

$$\forall n : nat, \forall l : (list\ A), (length\ l) = n\ \rightarrow\ (accs\ l)\ \rightarrow\ (acc_{<^{lex}_{|accs}}\ l)$$

Proof We proceed by induction on n.
Base Case $n = 0$ and then $l = nil$. From definition (9), nil cannot be compared with any list, and thus, by definition 4, it satisfies $acc_{<^{lex}_{|accs}}$.

Induction Step Let n be a natural number. Under the induction hypothesis **Hind**$_1$: $\forall l, (length\ l) = n\ \rightarrow\ (accs\ l)\ \rightarrow\ (acc_{<^{lex}_{|accs}}\ l)$
we have to prove this property for $n + 1$. A length-$(n + 1)$ list is of the form $h :: l$. Assuming that it satisfies predicate $accs$, we deduce that $(acc_<\ h)$ and $(accs\ l)$. Therefore we are led to prove a new goal of the form

$$\forall h : A,\ (acc_<\ h)\ \rightarrow\ (P\ h)$$

with $\quad P = \lambda h.\ \forall l, (length\ l) = n \rightarrow (accs\ l) \rightarrow (acc_{<^{lex}_{|accs}}\ l) \rightarrow (acc_{<^{lex}_{|accs}}\ h :: l)$
We proceed by induction on $(acc_<\ h)$. We have to prove $(P\ h)$ under the induction hypothesis: **Hind**$_2$: $\forall h' : A, h' < h\ \rightarrow\ (P\ h')$
Let us consider a list l, such that **H**$_0$: $(length\ l) = n$ and **H**$_1$: $(acc_{<^{lex}_{|accs}}\ l)$. We must prove the goal $(Q\ l)$ where Q is defined by :
$Q := \lambda l.(accs\ l) \rightarrow (acc_{<^{lex}_{|accs}}\ h :: l)$.
By applying principle (5) on **H**$_1$, we get the induction hypothesis:
Hind$_3$: $\forall l',\ (accs\ l')\ \rightarrow\ l' <^{lex} l\ \rightarrow\ (Q\ l')$.
Assume **H**$_2$: $(accs\ l)$. The goal is now $(acc_{<^{lex}_{|accs}}\ h :: l)$, that is, from definition (4), $\forall l', (accs\ l')\ \rightarrow\ l' <^{lex} h :: l\ \rightarrow\ (acc_{<^{lex}_{|accs}}\ l')$.
Let l' such that **H**$_3$: $(accs\ l')$ and **H**$_4$: $l' <^{lex} h :: l$. The goal is now $(acc_{<^{lex}_{|accs}}\ l')$.
From **H**$_4$, and by definition (9), two cases are considered:

- l' is of the form $l' = h :: l''$ with $l'' <^{lex} l$. Moreover, from **H**$_3$, we deduce that $(accs\ l'')$. Thus, we can conclude by **Hind**$_3$.
- l' is of the form $l' = h' :: l''$ with $h' < h$ and $(length\ l'') = (length\ l)$. From **Hind**$_2$, h' satisfies predicate P. But, $(length\ l'') = n$ from **H**$_0$. It follows from **H**$_3$ that $(accs\ l'')$, and thus we have $(acc_{<^{lex}_{|accs}}\ l'')$ from **Hind**$_1$. Therefore, all the premises of $(P\ h')$ are fulfilled, and so, we are done. □

Lemma 11 *For all sets A and for all binary relations $<$ on A, the multiset order $<^{mul}$ on the lists is a lifting.*

Proof The proof is not trivial. It relies on the fact that $<^{mul}$ is included in the transitive closure of a reduction relation $<^{red}$ defined by :

$$M \cup X <^{red} M \cup \{\{a\}\} \leftrightarrow \forall x \in X, x < a$$

Let us point out that the inverse inclusion holds as soon as $<$ is transitive. But this extra hypothesis is not required here. One can prove that all finite multisets whose elements are accessible for $<$ are accessible for $<^{red}$ [Nip98, Kop04], which is stronger than the $lifting$ property. It is well known that the accessibility for a relation implies the accessibility for its transitive closure, and therefore for the stronger relation $<^{mul}$. We refer the reader to [CD05] for more details on this lemma.

6 Related Work and Conclusion

As far as the well-foundedness of RPO is concerned, various non constructive proofs have been performed as in Dershowitz's paper [Der82]. Ferreira and Zantema in [FZ95] demonstrate several theorems related to the well-foundedness of first order term orderings. Their results are quite general and even complete with respect to the termination of the term rewriting systems in the case of finite signatures. They can be applied straightforwardly to RPO. But although their proofs do not rely on Kruskal's theorem, they are not constructive.

In a precursory attempt towards an elementary proof of the well-foundedness, Lescanne [Les82] introduces a decomposition order (DO) on the terms, that he proves to be equivalent to MPO. Then, he establishes the well-foundedness of DO assuming that the precedence relation on the functional symbols is total and well-founded. His proof is not really constructive, but it seems that with some efforts a constructive version, more intricate than ours, could be obtained. Moreover, Zorn's Lemma is required when the signature is infinite and the precedence non total, to embed it in a total one. However, this approach is interesting since it provides an efficient algorithm for comparing two terms.

In [JR99, JR03] Jouannaud and Rubio propose a constructive proof of termination of higher-order of recursive path ordering (horpo) by the Tait-Girard technique [GLT88] whose specialization to the case of first order terms leads to a proof of the well-foundedness of MPO by structural induction on terms as pointed out in [van01]. Our Coq proof relies on a simplification of this specialization, in the sense that it does not require the auxiliary lexicographic order on triples they use.

In [GL01], Goubault-Larrecq establishes a theorem the proof of which has been carried out in the Coq proof assistant. The result is general since it does not depend on the term structure, and therefore it applies to other algebras. The proof of the theorem is elementary. However, proving that it generalizes Ferreira and Zantema's results involves a non constructive step. Moreover, applying this theorem to RPO, and in particular, proving that hypothesis (iv) is satisfied is not simpler than our direct proof.

Dawson and Goré [DG04] prove a general theorem for establishing the well--foundedness of relations closed under context. This theorem has been machine-checked in Isabelle. Again, proving that the hypotheses of the theorem are satisfied may be difficult and requires in particular to find out an appropriate auxiliary relation to be proved well-founded (that may be non trivial). The authors apply it to various examples including LPO. The proof obtained in this last case seems reasonably easy and it would be interesting to compare this approach for RPO with a direct one as ours.

Let us also mention the work of Leclerc [Lec95] who carries out in Coq a termination proof of term rewriting systems with RPO. However, the proof is achieved without using the well-foundedness of RPO, which is not established, but rather an embedding of the rewrite relations into some well-founded ordering based on the Grzegorzcyk hierarchy of number theoretic functions.

We have given a constructive axiomatization of RPO in the CIC, and we have applied it to MPO, LPO and RPO with multiset and lexicographic status. We have strongly structured our approach and tried to use hypotheses as weak as possible when proving that RPO are well-founded monotonic strict orders. We aimed at giving a good insight into the properties, by simplifying at most their proofs and pinpointing their possible independence. This could be a good starting point towards the study of *horpo*, for instance. This work results in a Coq development available in the CoLor Library (http://color.loria.fr). Its modular architecture allows to enrich easily the model with other status such as, for instance, the pointwise order on the lists.

References

[CD05] Solange Coupet-Grimal and William Delobel. An Effective Proof of the Well-foundedness of the Multiset Path Ordering. In *Applicable Algebra in Engineering, Communication and Computing, AAECC*, submitted, 2005.

[Der82] Nachum Dershowitz. Orderings for Term Rewriting Systems. *Theoretical Computer Science*, 3(17):279–301, 1982.

[DG04] E.Jeremy Dawson and Rajeev Goré. A General Theorem on Termination of Rewriting. In *Computer Science Logic, CSL'04*, number 3210 in LNCS, pages 100–114. Springer-Verlag, 2004.

[FZ95] Maria.C.F. Ferreira and Hans Zantema. Well-foundedness of Term Orderings. In *4th International Workshop on Conditional Term Rewriting Systems (CTRS'94)*, number 968 in LNCS, pages 106–123. Springer-Verlag, 1995.

[GL01] Jean Goubault-Larrecq. Well-Founded Recursive Relations. In *15th Worshop on Computer Science Logic (CSL'01), Paris*, volume 2142 of *LNCS*, pages 484–497. Springer-Verlag, 2001.

[GLT88] Jean-Yves Girard, Yves Lafont, and Paul Taylor. *Proofs and Types*. Cambridge Tracts in Theoritical Computer, Science 7, 1988.

[JR99] Jean-Pierre Jouannaud and Albert Rubio. The Higher-Order Recursive Path Ordering. In *Proceedings of the 14th annual IEEE Symposium on Logic in Computer Science (LICS'99)*, pages 402–411, Trento, Italy, 1999.

[JR03] Jean-Pierre Jouannaud and Albert Rubio. Higher-Order Recursive Path Orderings a la carte. Technical report, http://www.lix.polytechnique.fr/ Labo/Jean-Pierre.Jouannaud/biblio.html, 2003.

[KL80] Sam Kamin and Jean-Jacques Levy. Two generalizations of the recursive path ordering. Technical report, University of Illinois, 1980.

[Kop04] Adam Koprowski. Well-foundedness of the Higher-Order Recursive Path Ordering in Coq. Master thesis, Free University of Amsterdam (The Netherlands) and Warsaw University (Poland), 2004.

[Kru60] J.B. Kruskal. Well-quasi-ordering, the tree theorem, and Vazsonyi's conjecture. *Trans. AMS*, 95:210–225, 1960.

[Lec95] François Leclerc. Termination Proof of Term Rewriting System with the Multiset Path Ordering. A Complete Development in the System Coq. In *TLCA*, pages 312–327, 1995.

[Les82] Pierre Lescanne. Some Properties of Decomposition Ordering, a Simplification Ordering to Prove Termination of Rewriting Systems. *R.A.I.R.O. Theoretical Informatics*, 14(4):331–347, 1982.

[Les83] Pierre Lescanne. Computer Experiments with the REVE Term Rewriting System Generator. *10th ACM Symposium on Principles of Programming Languages (POPL)*, pages 99–108, 1983.

[Nip98] Tobias Nipkow. An Inductive Proof of the Well-foundedness of the Multiset Order. Due to Wilfried Buchholz. Technical report, http://www4.informatik.tu-muenchen.de/~nipkow/misc/index.html, 1998.

[Tea04] The Coq Development Team. The Coq Proof Assistant Reference Manual – Version V8.0. Technical report, LogiCal Project-INRIA, 2004.

[van01] Femke van Raamsdonk. On termination of higher-order rewriting. In *Proceedings of the 12th International Conference on Rewriting Techniques and Applications (RTA'01)*, pages 261–275, Utrecht, The Netherlands, 2001.

Appendix : The Modular Architecture of the Coq Library

Figure 1 displays the modular architecture of the Coq library. It is composed of three main modules, each of them parameterized by a signature S:

- Module RPO_Hyps contains all the axioms, classified in three module types which are related respectively to the fixpoint definition (RPO_Axioms_Type), the hypotheses for the well-foundedness (RPO_Wf_Type), and those for the relation to be a monotonic strict partial order (RPO_MSO_Type). The last two inherit of the first one.
- Module RPO_Facts as a similar structure, and contains the proofs of the properties. Three sub-modules take as parameter a module of type one of the three module types defined in RPO_Hyps. They correspond to the proofs presented in sections 4.2, 4.3 and 4.4 respectively.

- In module *RPO_Model*, a model is built. The relation $<_{IRPO}$, with the two possible status, is defined. Two modules dedicated to the lexicographic order and the multiset order are loaded. They are parameterized by the base setoïd A which is then instantiated by the first order terms. Finally, one builds three modules that model the axiomatization since they have the correct module type. They correspond to the proofs of sections 5.1, 5.2, and 5.3 respectively.

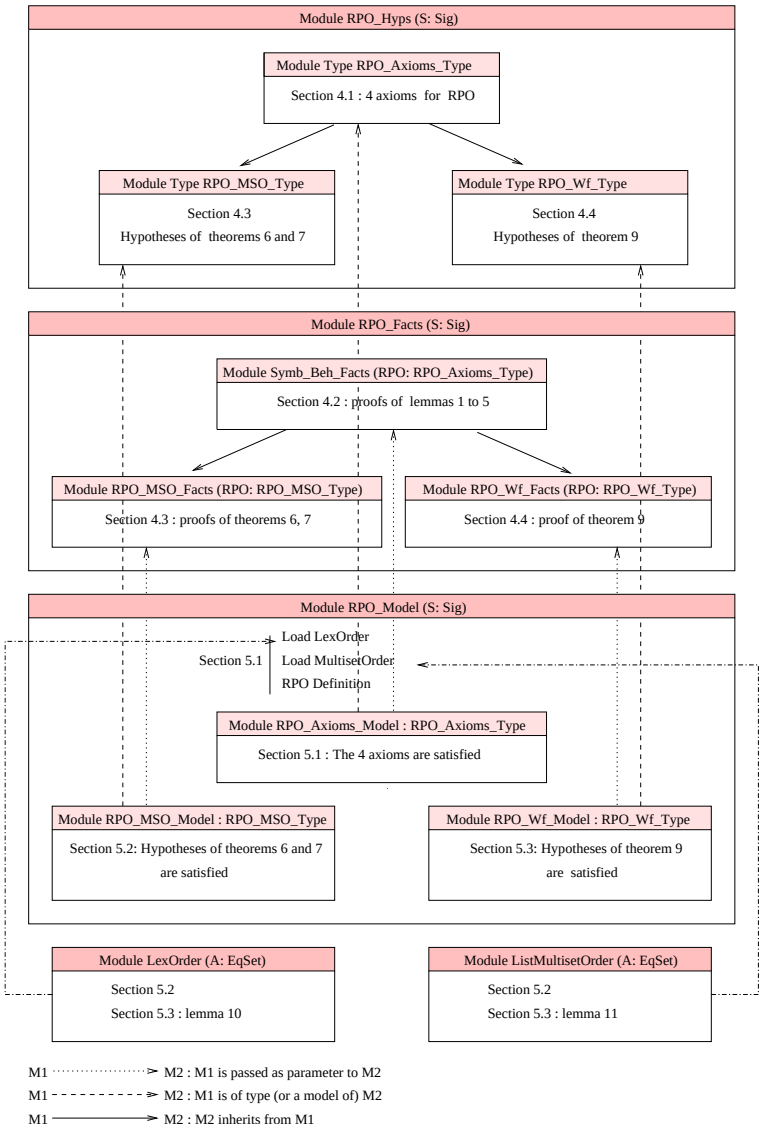

Figure 1: Architecture of the Coq Library