COMPARING AND PUTTING TOGETHER RECURSIVE PATH ORDERING, SIMPLIFICATION ORDERINGS AND NON-ASCENDING PROPERTY FOR TERMINATION PROOFS OF TERM REWRITING SYSTEMS

Alberto Pettorossi

IASI-CNR
Via Buonarroti 12
00185 ROMA (ITALY)

Computer Science Department
Edinburgh University
EDINBURGH EH9 3JZ (Scotland)

Summary

Abstract

We give a sufficient condition for proving strong termination in Combinatory Logic and Rewriting Systems which solves an open problem [Böh 77]. We also compare, in the context of general rewriting systems, the power of that condition and other known methods, as the recursive path orderings and simplification orderings, presenting original results.

A new technique for proving strong termination, called Diagram of Matchings, is also introduced. In many cases it allows to combine together the strength of various methods of proof.

1. Introduction

During the last few years many interesting papers have been published on the problem of proving termination of rewriting systems [HuO 80]. Back in 1975, while studying strong termination in Combinatory Logic systems, we introduced the Non-Ascending property (NA property, for short) [Pet 75]. In this paper we prove that NA property without restrictions implies strong termination. Our proof is based on the recursive path ordering method [Pla 78] and it does not require any generalization of such method as suggested by [KaL 80].

Since the strong termination problem is undecidable, no general method exists for its solution. What one can do is to compare the various methods and analyze their relative power. We do this in section 4 and we compare the NA property, the recursive path orderings and the simplification orderings. In some cases the NA property shows its strength for proving strong termination of rewriting systems, because it avoids the need of looking for complicated (quasi) orderings, as required by other methods [Der 79c]. Indeed in section 5., using NA property, we will give a short proof of the strong termination of Example (F) in [Der 79c] (there the proof is based on the invention of a quite clever ordering).

We also present a method, called Diagram of Matchings, for putting together the strength of single techniques of proving strong termination. This method is useful, because the undecidability constraint already mentioned, implies that no single technique is uniformly satisfactory.

2. Combinatory Logic Rewriting Systems and preliminary definitions

In this section we will recall some definitions concerning a particular rewriting system, called Weak Combinatory Logic (WCL, for short) [Cur 58], in which the Non-Ascending property for proving short termination can be best introduced.
The alphabet of WCL is: I,K,S constants or basic combinators; (,) parentheses for building terms and x,y,t,... variables which range over terms.
Terms are defined as follows:
i) a constant or a variable is an atomic term;
ii) an application of two terms x_1 and x_2, denoted by $(x_1\ x_2)$, is a term. Thus terms in WCL look like binary trees.

We will assume left associativity so that $x_1x_2...x_n$ stands for $(...(x_1x_2)...x_n)$.
Combinators are terms without variables.

In WCL we have also a binary relation $\rightarrow$ on the set of terms, which we will call reduction. It is defined by the following reduction axioms:

1. $Ix_1 \rightarrow x_1$ (reduction axiom for I)
2. $Kx_1x_2 \rightarrow x_1$ (reduction axiom for K)
3. $Sx_1x_2x_3 \rightarrow x_1x_3(x_2x_3)$ (reduction axiom for S)

We assume that reflexivity does not hold for $\rightarrow$.
Inference rules of WCL are:

4. Transitivity holds for $\rightarrow$.

5.1 If $t_1 \rightarrow t_2$ then $t_ot_1 \rightarrow t_ot_2$ (right monotony)
5.2 If $t_1 \rightarrow t_2$ then $t_1t_o \rightarrow t_2t_o$ (left monotony)
$\equiv$ denotes syntactical identity of two terms.

We will write $t \underset{n}{\rightarrow} v$ for showing that the term v is obtained from the term t by using n times the reduction axioms. We can introduce in WCL some other constants (or basic combinators) defining them either in terms of applications of I,K and S or by giving the corresponding reduction axiom. For example, we can introduce a constant B by giving either $B \equiv S(KS)K$ or $Bx_1x_2x_3 \rightarrow x_1(x_2x_3)$. We refer to [Cur 58] for the notion of redex, normal form, etc.

Definition 1. The order of a basic combinator X is the least integer m such that every reduction of $Xx_1 \ldots x_m y_1 \ldots y_n$ is obtained by applying the left monotony rule to the reduction of $Xx_1 \ldots x_m$.

We also introduce some more definitions. We say that a combinator t has strong normal form when all reduction strategies lead it to its normal form (usually t is also said to have strong normalization property).

A subbase B is a non-empty (possibly infinite) set of basic combinators, i.e. $B = \{X_1, \ldots, X_n\}$.

The applicative closure B^+ of a subbase B is the set of all finite applicative combinations of the basic combinators in B.

The set S_t of subterms of the term t is defined as follows:

i) if t is a basic combinator then $S_t = \{t\}$

ii) if $t \equiv (t_1t_2)$ then $S_t = S_{t_1} \cup S_{t_2} \cup \{t\}$

The set of proper subterms of t is $S_t - \{t\}$.

We say that the basic combinator X with reduction axiom $Xx_1 \ldots x_n \rightarrow v$ is a proper combinator iff v is an applicative combination of variables in $\{x_1, \ldots, x_n\}$.

Given a term t, we define the corresponding marked term marked(t) as follows:

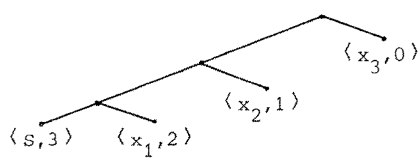
marked(t) $\equiv$ mark(t,0) where

mark(t,n) = $\langle t,n \rangle$	if t is a basic combinator or a variable
= (mark(t_1,n+1) mark(t_2,n))	if $t \equiv (t_1t_2)$

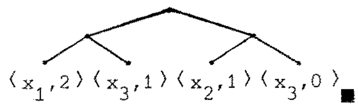
A marked term can be represented in the obvious way as a binary tree, whose leaves are labelled by pairs.

In a marked term t an occurrence v of an atomic subterm is associated with an integer, which we will call the c-number (short for copy-number), of that atomic subterm occurrence. It will be denoted by cn(v,t). The c-number of a leaf of a term t is the number of left choices one has to make in going from the root of the tree corresponding to t to the considered leaf.

Example 1. marked($Sx_1x_2x_3$) looks like:



marked($x_1x_3(x_2x_3)$) looks like:



■

Now we will introduce the notion of rewriting system as a generalization of WCL. Let F be a (finite) set of operators, i.e. symbols with arity. T(F) is the set of (ground) terms over F.

Definition 2. A rewriting system Σ is a finite set of axiom schemata where variables range over T(F). The i-th axiom schema is of the form: $\ell_i(\vec{x}) \to r_i(\vec{x})$ where $\vec{x}$ denotes a vector of variables and $\mathrm{var}(r_i(\vec{x})) \subseteq \mathrm{var}(\ell_i(\vec{x}))$.
A substitution $\vartheta(x)$ is ground iff $\vartheta(x)$ associates to the variable x a term in T(F).
Given a term t_1 we can obtain a term t_2 by applying once the i-th rule of Σ, and we write $t_1 \xrightarrow[1(i)]{} t_2$, iff:

- t_1 or one of its subterms matches $\ell_i(\vec{x})$, according to the (ground) substitution $\vartheta(\vec{x})$, and
- t_2 is obtained from t_1 by replacing t_1 or the considered subterm of t_1 with $r_i(\vartheta(\vec{x}))$.

Notice that the rewriting process is highly non deterministic because more than one rewriting rule can be applied and a rule can be applied with respect to one or more (sub)terms of t_1.
Example 2. Given $\Sigma = \{\neg\neg x \to x;\ \neg(x \wedge y) \to \neg x \vee \neg y;\ \neg(x \vee y) \to \neg x \wedge \neg y\}$, and $F = \{\vee, \wedge, \neg, A\}$ where $\vee, \wedge, \neg$ and A have arity 2,2,1 and 0 respectively, $t_1 \equiv \neg(\neg\neg A \vee \neg A)$ can be rewriten as $\neg\neg\neg A \wedge \neg\neg A$ or as $\neg(A \vee \neg A)$.

Definition 3. A rewriting system Σ is strongly terminating w.r.t. the set of term T(F) iff $\forall t \in T(F)$ there is no infinite sequence of terms $t_1 \to t_2 \to \ldots \to t_n \to \ldots$ s.t. $t_1 \equiv t$ and $\forall K \exists i$ s.t. $t_k \xrightarrow[1(i)]{} t_{k+1}$.

3. The Non-Ascending Property as a sufficient condition for strong termination

Definition 4. We say that a proper combinator X with reduction axiom $Xx_1x_2\ldots x_n \to v$ has Non-Ascending property (NA property, for short) iff $\forall i$ for $1 \le i \le n$ if $\langle x_i, p\rangle$ occurs in marked($Xx_1\ldots x_n$) and $\langle x_i, q\rangle$ occurs in marked(v) then $p \ge q$.

Notice that in marked($Xx_1\ldots x_n$) x_i is associated with $p = n-i$, i.e. $\mathrm{cn}(x_i, \mathrm{marked}(Xx_1\ldots x_n)) = n-i$.
Example 3. If $Xx_1x_2 \to x_1(x_1x_2)$ X has NA property.
If $Xx_1x_2x_3x_4 \to x_1(x_1(x_1x_2))(x_2(x_2(x_2x_3))(x_2x_3x_4)$ X has NA property. S does not have NA property, because $\langle x_3, 0\rangle$ occurs in marked($Sx_1x_2x_3$) and $\langle x_3, 1\rangle$ occurs in marked($x_1x_3(x_2x_3)$). ∎
In order to prove that the NA property is a sufficient condition for strong termination we will use the recursive path ordering method [Pla 78]. We will follow Dershowitz's notation [Der 79c].

Let us first recall some preliminary definitions and theorems.
Given any set A, we will denote by $\mathcal{M}(A)$ the set of multisets built out of elements in A. For instance, if A is the set of natural numbers, $\{1,3,2,1,3\} \in \mathcal{M}(A)$.

Definition 5. Given an ordered set (A, >), we can define an ordering >> for the multisets $\mathcal{M}(A)$ as follows: $\forall M,N \in \mathcal{M}(A)$ $M >> N$ iff $M = Z \dot{\cup} A$ and $N = Z \dot{\cup} B$ and $\forall b \in B$ $\exists a \in A$ s.t. $a > b$, where $\dot{\cup}$ denotes disjoint union of multisets. Notice that A cannot be empty.

For instance: $\{5,5,4,1\} \gg \{5,5,3,2,3\}$ where $Z = \{5,5\}$, $A = \{4,1\}$ and $B = \{3,2,3\}$.

Definition 6. The ordered set $(A, >)$ is said to be well-founded iff there is no infinite descending sequence of elements in A such as: $a_1 > a_2 > \ldots > a_n > \ldots$

Theorem 1. [Der 79a] i) If $>$ is irreflexive and transitive then $\gg$ is irreflexive and transitive. ii) $>$ is well-founded iff $\gg$ is well-founded.

Let $\cdot\!>$ be an irreflexive and transitive ordering on a set of operators F with fixed arity.

Definition 7. The recursive path ordering $>$ over the set T(F) of terms over F is recursively defined as follows:

$$s \equiv f(s_1,\ldots,s_m) > g(t_1,\ldots,t_n) \equiv t$$

iff $s \neq t$ and [either ($f \cdot\!> g$ and $s > t_i$ for all i) (RPO.1)

or ($f \equiv g$ and $\{s_1,\ldots,s_m\} \gg \{t_1,\ldots,t_n\}$ with $m=n>0$) (RPO.2)

or ($s_i \geq t$ for some i, $1 \leq i \leq m$)] (RPO.3)

where: i) $s \equiv t$ iff they are exactly the same term or operator

ii) $s \geq t$ iff ($s > t$ or $s = t$)

iii) $s = t$ iff ($f \equiv g$ and $\langle t_1,\ldots,t_m \rangle$ is a permutation of $\langle s_1,\ldots,s_m \rangle$ where m is the arity of f and g)

iv) $\gg$ is the multiset extension of the ordering $>$.

Theorem 2. [Der 79c]. $\cdot\!>$ is well-founded iff $>$ is well-founded.

Definition 8. An irreflexive and transitive ordering $>$ over a set of terms T(F) is a simplification ordering if it satisfies the following properties:

i) $t > t'$ implies $f(\ldots,t,\ldots) > f(\ldots,t',\ldots)$ (monotonicity)

ii) $f(\ldots,t,\ldots) > t$ (subterm)

for any $t,t' \in T(F)$ and any $f \in F$.

Theorem 3 [Der 79c]. A recursive path ordering over a set of terms T(F) is a simplification ordering.

The interest of recursive path orderings and simplification orderings relies on the following theorem:

Theorem 4 [Der 79c]. Given a rewriting system $\Sigma = \{\ell_i(\vec{x}) \rightarrow r_i(\vec{x}) \mid 1 \leq i \leq p\}$ if there exists a simplification ordering $>$ s.t. for any ground substitution $\vartheta(\vec{x})$, $\ell_i(\vartheta(\vec{x})) > r_i(\vartheta(\vec{x}))$ then Σ is strongly terminating.

Analogously to Theorem 4., the NA property also gives a sufficient condition for strong termination. For showing this fact, we need to prove first some preliminary lemmas.

In what follows, unless otherwise stated, we will consider X to be a proper combinator whose order is n.

Fact 1. X has NA property iff its reduction axiom can be written as follows [Lév 80]:

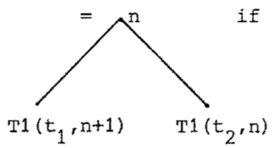
$Xx_1 \ldots x_n \to t_n$ where $t_n \in \mathbf{T}_n = \{t_{n-1}\, u_n \mid t_{n-1} \in \mathbf{T}_{n-1}\ ,\ u_n \in \mathbf{T}_n\} \cup \{x_1,\ldots,x_n\}$ and $\mathbf{T}_o = \{\ \}$.

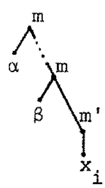
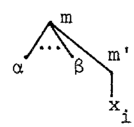
Proof. By induction on n and on the size of t_n. ∎

For each $t \in \{X,x_1,\ldots\}^+$, we define the following tree Tr(t), whose leaves are the atomic terms occurring in t and all other nodes are associated with primed or unprimed natural numbers (those numbers are considered as operators with arity 2 if unprimed, 1 if primed).

Tr(t) = Modify (T1(t,0))

T1(t,n) =  if t is atomic

=  if $t \equiv t_1 t_2$

where Modify changes any maximal subtree of the form  into 

Therefore the set of operators we used for the trees in {Tr(t)} is:

$$F = \mathbf{N} \cup \mathbf{N}' \cup \{x_1,\ldots,x_n\} \cup \{X\}$$

where **N** and **N'** are respectively the set of unprimed and primed natural numbers. Due to the Modify operation, the operators in F should be considered to have variable arity, but this does not create problems [Der 79a]. We assume the following irreflexive and transitive order among the elements in F:

a •> b iff a is greater than b in the usual order among natural numbers without considering the prime mark.

All other pairs of elements of F are unrelated.

Example 4. 5 •> 3 •> 2' and 5' •> 3' •> 2.

Let us denote by > the r.p.o. defined in {Tr(t)} starting from the ordering •> .
Since the ordering •> is obviously well-founded, > also is well-founded ([Der 79c]).

Given the terms (or trees) t, t_1 and t_2 , s.t. t_1 is a particular occurrence of a subterm (or subtree) of t, we will denote by $t[t_1 \leftarrow t_2]$ the result of the replacing t_1 by t_2 in t.

$t[\vec{t}_1 \leftarrow \vec{t}_2]$ denotes the extension of the replacing operation to a vector of occurrences. It is defined in a componentwise way.

Lemma 1. (Context Lemma 1). Given t, t_1 and $t_2 \in \{X\}^+$ s.t. t_1 is an occurrence of a subterm of t, if $Tr(t_1) > Tr(t_2)$ then $Tr(t) > Tr(t[t_1 \leftarrow t_2])$.

Proof. Immediate by definition of r.p.o. ■

Lemma 2. If $Xx_1 \ldots x_n \rightarrow t_n$ and X has NA property then for any i, $1 \le i \le n$, and for any $n \ge 1$
$Tr(Xx_1 \ldots x_n) > Tr(t_n)$.

Proof. (i) Suppose $t_n \equiv x_n$. Since $Tr(t_n) = \begin{array}{c} 0' \\ | \\ x_n \end{array}$ is a subterm of $Tr(Xx_1 \ldots x_n)$ the thesis follows by RPO.3.

(ii) Suppose $t_n \equiv x_i$ for $i \neq n$. Since a subterm of $Tr(Xx_1 \ldots x_n)$ is $\begin{array}{c} m' \\ | \\ x_i \end{array}$ for $m' > 0'$, the thesis follows by RPO.3.

(iii) Suppose $t_n = t^1_{n-1}(t^2_{n-1}(\ldots x_i))$. We need to prove that:

$$\{Tr(Xx_1 \ldots x_{n-1}), \begin{array}{c} 0' \\ | \\ x_n \end{array}\} \gg \{Tr(t^1_{n-1}), Tr(t^2_{n-1}), \ldots, \begin{array}{c} 0' \\ | \\ x_i \end{array}\}.$$

This is true by induction hypothesis on n and because:

if $i = n$ then $\begin{array}{c} 0' \\ | \\ x_n \end{array} = \begin{array}{c} 0' \\ | \\ x_i \end{array}$; if $i \neq n$ then $\begin{array}{c} m' \\ | \\ x_i \end{array}$ with $m' > 0'$ occurs in $Tr(Xx_1 \ldots x_{n-1})$. ■

Lemma 3. (Context Lemma 2). Given the terms t_1, t_2 and r, let us consider the occurrences o_1 and o_2 of the same atomic subterm respectively in t_1 and t_2.
If $Tr(t_1) > Tr(t_2)$ then $Tr(t_1[o_1 \leftarrow r]) > Tr(t_2[o_2 \leftarrow r])$.

Proof. Immediate. ■

Lemma 4. Given $t \equiv Xx_1 \ldots x_n$, $t' \in \mathbf{T}_n$ and $\vec{r}$ a vector of terms in $\{X\}^+$ we have
$Tr(t[\vec{x} \leftarrow \vec{r}] > Tr(t'[\vec{x} \leftarrow \vec{r}])$.

Proof. By induction on the length of the vector of variables $\vec{x}$. ■

Let us consider the generic component, say x_j, of $\vec{x}$. By Lemma 2, $Tr(t) > Tr(t')$.
If $t' = t^1_{n-1}(t^2_{n-1}(\ldots x_i))$ then $\forall k$ in t^k_{n-1} there is only one occurrence of the variable x_{n-1}, because $t' \in \mathbf{T}_n$ (this can easily be proved by structural induction on t'). Therefore one can derive $Tr(t[\vec{x} \leftarrow \vec{r}]) > Tr(t'[\vec{x} \leftarrow \vec{r}])$ by applying Lemma 3 for the occurrences of the variable x_j when comparing $Tr(Xx_1 \ldots x_{n-1}[x_j \leftarrow r_j])$ with $Tr(t^k_{n-1}[x_j \leftarrow r_j])$ for any k. ■

Theorem 6. (The NA Theorem). $\forall t, t' \in \{X\}^+$ if X has NA property and $t \rightarrow t'$ then $Tr(t) > Tr(t')$. Thus $\forall t \in \{X\}^+$ has strong normal form.

Proof. Immediate by Context Lemma 1 and Lemma 4. ■

Example 5. Given $+$ s.t. $+\, x_1x_2x_3x_4 \rightarrow x_1x_3(x_2x_3x_4)$, any term $t \in \{+\}^+$ has strong normal form.

Theorem 7. Given a subbase $B = \{X_1, X_2, \ldots\}$ s.t. $\forall i$ X_i has NA property, $\forall t \in B^+$ t has strong normal form.

Proof. Immediate extension of Theorem 6. ■

Example 6. Given X and Y s.t. $Xx_1x_2x_3x_4 \rightarrow x_1(x_1x_2)(x_1x_2x_3)(x_1x_2(x_1x_2x_3)x_4)$ and $Yx_1x_2x_3 \rightarrow x_1(x_1(x_1(x_1(x_1x_2))))(x_1(x_1x_2)(x_1x_2x_3))$, $\forall t \in \{X,Y\}^+$ t has strong normal form. ■

The proof of the main lemma 2 we have given, is based on the r.p.o. method by [Pla 78]. It is possible to give another proof of the NA theorem [Lév 80], but it relies on a generalization of the r.p.o. method [KaL 80], which consists in replacing (RPO.2) by the following:

$$f \equiv g \text{ and } \langle s_1,\ldots,s_m\rangle >>_{\text{lexicographically}} \langle t_1,\ldots,t_n\rangle \text{ with } m = n > 0 \text{ and } \forall i\ s > t_i \qquad \text{(RPO.2')}$$

4. Comparing the Non-Ascending property, Recursive Path Orderings and Simplification Orderings

We will show a general result concerning the relative power of the two Recursive Path Orderings methods we mentioned in the previous section, namely:

1st method (with multiset order): (RPO.1) + (RPO.2) + (RPO.3), which we will denote by RPO.MS, and

2nd method (with bounded lexicographic order): (RPO.1) + (RPO.2') + (RPO.3), which we will denote by RPO.BL.

In proving termination we will say that a method M1 is not less powerful than another method M2, and we write M1 $\trianglerighteq$ M2, if for any proof using M2 there exists a proof using M1.

Theorem 8. RPO.MS $\trianglerighteq$ RPO.BL.

Proof. It is enough to show how a bounded lexicographic ordering can be translated into a multiset ordering. Let us consider the lexicographic order to be left-to-right, i.e. $\langle a,b\rangle > \langle c,d\rangle$ iff $(a > c$ or $(a = c$ and $b > d))$.

We can transform the pair $\langle a,b\rangle$ into the set $\{\langle 1,a\rangle,\langle 0,b\rangle\}$. Then it is easy to show that $\langle a,b\rangle > \langle c,d\rangle$ iff $\{\langle 1,a\rangle,\langle 0,b\rangle\} >> \{\langle 1,c\rangle,\langle 0,d\rangle\}$ where $\langle n,m\rangle > \langle n',m'\rangle$ in the usual lexicographic order.

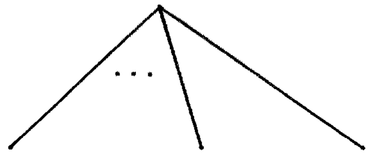
If, instead of pairs, we have n-tuples, we must encode them into pairs, i.e. $\langle a_1,\ldots,a_n\rangle = \langle a_1, \ \ldots \ \langle a_{n-1},a_n\rangle\ldots\rangle$. Therefore, for instance, $\langle a,b,c\rangle$ becomes the set of pairs $\{\langle 1,a\rangle,\langle 0,\langle 1,b\rangle\rangle,\langle 0,\langle 0,c\rangle\rangle\}$, or equivalently the set $\{\langle 2,a\rangle,\langle 1,b\rangle,\langle 0,c\rangle\}$.

In order to complete our proof we have to show how to encode in a multiset ordering the fact that the lexicographic ordering is bounded. We need to show that $t_1 > t_2$, where $t_1,t_2 \in T(F)$, according to (RPO.BL), iff $Tr1(t_1) > Tr2(t_2)$ according to (RPO.MS) for some transformations Tr1 and Tr2. They are:

$Tr1(t_1) = T1(0,t_1)$ <u>where</u>

$T1(n,t_1) = \langle n,t_1 \rangle$ if t_1 is atomic

$= \langle n,f \rangle$ if $t_1 \equiv f(t_{11},\ldots,t_{1k})$

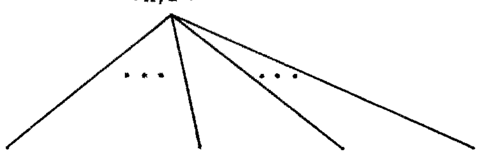


$T1(n+k-1,t_{11})$ $T1(n,t_{1n})$ $\langle *,t_1 \rangle$ where $*$ is an extra unrelated element of F,and

$Tr2(t_2) = T2(0,t_2)$ <u>where</u>

$T2(n,t_2) = \langle n,t_2 \rangle$ if t_2 is atomic

$= \langle n,f \rangle$ if $t_2 \equiv f(t_{21},\ldots,t_{2k})$



$T2(n+k-1,t_{21})$ $T2(n,t_{2k})$ $\langle *,t_{21} \rangle$ $\langle *,t_{2k} \rangle$

Notice that in comparing for instance $\langle *,t_1 \rangle$ with $\langle *,t_{2j} \rangle$, we have to compare t_1 with t_{2j}, and this must be done using again the bounded lexicographic ordering. But we have to use RPO.MS and not RPO.BL. This forces us to construct $Tr1(t_1)$ again (notice that we already constructed $Tr1(t_1)$ when comparing t_1 and t_2). Nevertheless this process of constructing $Tr1(t_1)$ will eventually terminate, by induction on the pair of sizes $\langle \|t_1\|, \|t_2\| \rangle$. ■

<u>Example 7</u>. Ackermann function definition.

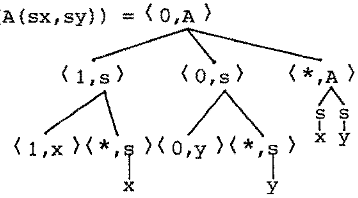
1. $A(0,x) \rightarrow sx$
2. $A(sx,0) \rightarrow A(x,s0)$
3. $A(sx,sy) \rightarrow A(x,A(sx,y))$

No <u>direct</u> proof of strong termination is possible using RPO.MS, because it is not the case that $\{sx,sy\} >> \{x,A(sx,y)\}$ [KaL 80]. But we can have an indirect proof using Theorem 8. Let us consider $F = \{A,s,0\} \cup \mathbf{N} \cup \{*\}$, where

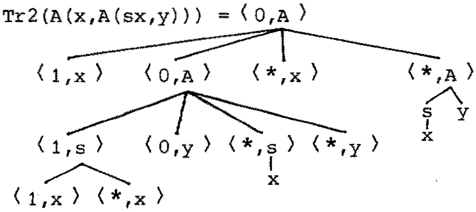
$A \bullet\!> s \bullet\!> 0$, $m \bullet\!> n$ if $m,n \in \mathbf{N}$ and $m > n$ in the usual ordering for $\mathbf{N}$.

The only difficult rule is the third one. We have:

$Tr1(A(sx,sy)) = \langle 0,A \rangle$



$Tr2(A(x,A(sx,y))) = \langle 0,A \rangle$



and we can easily prove $Tr1(A(sx,sy)) > Tr2(A(x,A(sx,y)))$, using RPO.MS and subterm induction on the r.h.s. of equation 3. ∎

Now we would like to compare the power of the recursive path ordering methods, which are particular simplification orderings, with the power of the NA property in proving strong termination in WCL. For simplicity reasons we will present our results when terms are elements of $\mathcal{B}^+ = \{X\}^+$, where X is a proper combinator with NA property, but the results are valid also when $\mathcal{B}^+ = \{X_1, X_2, \ldots\}^+$ and X_i is a proper combinator with NA property for any i.

Let us first present the (RPO.MS) method and the (RPO.BL) method, in the case of <u>W</u>eak <u>C</u>ombinatory <u>L</u>ogic with proper combinators (WCLp, for short).

We consider the set of operators $F = \{X, a, x_1, x_2, \ldots\}$ whose elements have arity 0, except a, which denotes application of terms and has arity 2. We assume $(t_1 t_2) \equiv a(t_1, t_2)$. We suppose $a \bullet\!> X$, and no other pairs are related under $\bullet\!>$. Notice that $\forall t \in T(F)$ if $x_i \notin S_t$ then t and x_i are not related by $>$. RPO.1 reduces to: $s > t$ iff $s \equiv a(s_1, s_2)$ and $t \equiv X$.
Given $s \equiv a(s_1, s_2) \equiv a((Xx_1 \ldots x_{n-1}), x_n)$ and $t \equiv a(t_1, t_2) \equiv a(u_{n-1}, u_n)$ where $u_{n-1} \in \mathbf{T}_{n-1}$ and $u_n \in \mathbf{T}_n$ we have that: RPO.2 reduces to $\{s_1, s_2\} >> \{t_1, t_2\}$, i.e. $\{Xx_1 \ldots x_{n-1}, x_n\} >> \{u_{n-1}, u_n\}$ and RPO.2' reduces to $[(s_1 > t_1)$ <u>or</u> $(s_1 = t_1$ <u>and</u> $s_2 > t_2)]$ <u>and</u> $s > t_1$ <u>and</u> $s > t_2$, i.e. $(s_1 > t_1$ <u>and</u> $s > t_2)$ <u>or</u> $(s_1 = t_1$ <u>and</u> $s_2 > t_2)$. Since $s_2 \equiv x_n$, RPO.2' is equivalent to $(Xx_1 \ldots x_{n-1} > u_{n-1}$ <u>and</u> $Xx_1 \ldots x_{n-1} x_n > u_n)$
RPO.3 reduces to $s_i \geq t$ for $i = 1,2$.

<u>Fact 2</u>. For WCLp, RPO.MS and NA property are incomparable, i.e. there are cases in which one can prove termination using RPO.MS, but not NA property and viceversa. (This fact holds when no previous isomorphic transformations of the terms to be compared are done).

<u>Proof</u>. i) RPO.MS $\trianglerighteq$ NA property. Consider $Xx_1x_2x_3 \rightarrow x_3x_2$.
NA does not hold, but $\{Xx_1x_2, x_3\} >> \{x_3, x_2\}$
ii) NA property $\trianglerighteq$ RPO.MS. Consider $Xx_1x_2x_3 \rightarrow x_1x_2(x_1x_2x_3)$.
$\{Xx_1x_2, x_3\} \not>\!\!> \{x_1x_2, x_1x_2x_3\}$, because $Xx_1x_2 \not> x_1x_2x_3$. ∎

<u>Fact 3</u>. For WCLp, RPO.BL and NA property are equivalent, i.e. RPO.BL $\trianglerighteq$ NA property and NA property $\trianglerighteq$ RPO.BL.

<u>Proof</u>. The proof is done by structural induction on t_n in $Xx_1 \ldots x_n \rightarrow t_n$. If $t_n \equiv x_i$ for $i = 1,2,\ldots,n$ then obviously RPO.BL and NA property are equivalent. If $t_n \equiv t_{n-1}u_n$ we already deduced that (RPO.2') is $(Xx_1 \ldots x_{n-1} > t_{n-1}$ <u>and</u> $Xx_1 \ldots x_{n-1}x_n > u_n)$. These two conditions correspond to the reductions axioms of two constants, say Y and Z, with NA property. In fact, since X has NA property, $t_n \in \mathbf{T}_n$ and by Fact 1 we have:
$t_{n-1} \in \mathbf{T}_{n-1}$, i.e. Y s.t. $Yx_1 \ldots x_{n-1} \rightarrow t_{n-1}$ has NA property, and
$u_n \in \mathbf{T}_n$, i.e. Z s.t. $Zx_1 \ldots x_n \rightarrow u_n$ has NA property.

In order to complete the prove we have to show that RPO.3 never applies. This is obvious because all non atomic terms have the same topmost operator a. ∎

5. Using various orderings together: the method of the Diagram of Matchings

Most of the rewriting systems one encounters, are not suitable for the direct application of the NA property, because they do not correspond to reduction of proper combinators. When dealing with non-proper combinators, the NA property alone is not longer a sufficient condition for strong termination, as shown by the following example.

Example 8. $\neg x \to \neg x$ is not terminating. NA property holds because the l.h.s. is equal to the r.h.s.

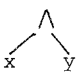
Therefore a modification of the NA property is necessary for coping with non-proper combinators or generic rewriting rules.

Let us first generalize the notion of c-number. This is done by a suitable definition of marked(t) for any term t, so that we can associate with operators or variables in t, suitable values in a well-founded domain. We will state without proofs the following theorems:

Theorem 9. If for any rewriting rule each variable strictly decreases the value of its c-number in a well-founded domain, then strong termination is guaranteed.

Theorem 10. If for any rewriting rule there is no duplication of variables and the multiset of the c-numbers associated with the constants decreases in a well-founded domain, then strong termination is guaranteed.

Example 9. $(x \wedge y) \wedge z \to x \wedge x$ is strongly terminating by Theorem 9, using as c-number the distance from the top of the tree obtained by representing $x \wedge y$ as



. In fact cn(x,marked($(x \wedge y) \wedge z$)) = 2 and cn(x,marked($x \wedge x$)) = 1 for each occurrence of x in $x \wedge x$. ∎

Example 10. [Der 79c]. $(x \wedge y) \wedge z \to x \wedge (y \wedge z)$ where the variables range over T(F) and F = $\{\wedge, A\}$. $\wedge$ and A have arity 2 and 0 respectively.
Choosing the prefix representation we have:
$\wedge \wedge xyz \to \wedge x \wedge yz$. Defining the c-number for each $\wedge$ as the number of A's occurring to its right, the sum of all c-numbers of $\wedge$'s decreases at each contraction. Strong termination is guaranteed by Theorem 10. ∎

Now we will present, through some examples, a new method for proving strong termination of rewriting systems. In various cases it allows to combine together the strength of many different methods of proof. The central idea is the analysis of the way in which, during the rewriting process, new redexes are generated, because termination is guaranteed if it is impossible to have an infinite generation of redexes. For this purpose we will define a graph, called Diagram of Matchings, and for proving termination we have to show that no infinite looping is possible in each strongly connected component of such a diagram. For each component we can use a different method of proof and this will provide us a way of putting together the strength of various methods. The following theorem gives the necessary justification of our approach. Let F be a set of operators with fixed arities and T(F) be the set of terms over F. Let $\Sigma = \{\ell_i(\vec{x}) \rightarrow r_i(\vec{x})\}$ be a rewriting system and t be a term in T(F).

We will denote by $R(t)$ the (multi)set of all redexes in t.

Theorem 11. If there exists a well-founded order > and a measure M on subsets of T(F) s.t. $\forall i\ \forall t,v \in T(F)$ s.t. $t \xrightarrow[1(i)]{} v$, $M(R(t)) > M(R(v))$ holds, then Σ is strongly terminating.

Proof. Since > is a well-founded ordering, no infinite descending sequences exist, and therefore strong termination is guaranteed.

The method, called Diagram of Matchings, consists in three steps:

1st Step: choice of an opportune algebraic representation of the terms in T(F), so that the structure of the redexes and the contexts in which they occur are made explicit. The invention of the algebraic representation effects also the measure M one has to discover for proving termination.

2nd Step: derivation of the Diagram of Matchings. A node in that diagram stands for a rewriting rule, and an edge, say $i \longrightarrow j$ indicates that after the contraction of a redex of the rule i (an i-redex for short), a j-redex can be created. Such a new redex can only be derived because of a matching of the r.h.s. of the i-th rule with the l.h.s. of the j-th rule. Therefore the generation of new redexes is captured by analysing the matchings of r.h.s. and l.h.s. of the rules.
Given a term t, we may consider that for each redex of t there is a contracting agent acting in the corresponding node of the diagram. At each contraction step the agent which performs it, so to speak, dies and creates one or more contracting agents in the nodes following the one where it was (in general it may also kill other agents in other nodes as well). Therefore the reduction process can be considered as a path in the tree (or the forest, if at the beginning the redexes are more than one) which is the unfolding of the Diagram of Matchings.

3rd Step: proof of finiteness of the unfolding forest. This can be done by showing that, for each tree of that forest, each branch has finite length and the number of branches is finite. Since the Diagram of Matchings is a directed graph and, in general, not strongly connected, we can prove that the unfolding forest is finite by using Theorem 11 with different orderings and measures for each strongly connected subdiagrams.

Example 11. [Der 79a].
a. $ffx \rightarrow fgfx$ where $F = \{f,g,A\}$. f and g have arity 1 and A has arity 0.
Since ffx can be embedded into fgfx no simplification ordering exists by which strong

termination can be proved [Der 79a pag. 213].

1st proof of termination. There is a bijection between T(F) and T(F') where $F' = \{|\} \cup F$ where $|$ has arity 1. It is defined by inserting $|$ between any two adjacent f's. Thus $|$ denotes the presence of an a-redex. The given rewriting system is strongly terminating iff $f|fx \to fgfx$ is strongly terminating. Since at each contraction step the number of $|$'s decreases, the latter system is strongly terminating. In this case the measure M mentioned in Theorem 11, by which we proved termination, is the cardinality of the set of redexes.

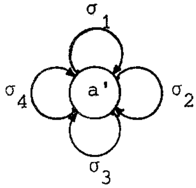
2nd proof of termination. We will use the Diagram of Matchings.
Given any term in T(F) we can consider the string of f's and g's to the left of the terminating A as a word of the free monoid Mon over F-{A}. Therefore (a) is strongly terminating iff the following rewriting system is strongly terminating:
a'. $x_i ffx_j \to x_i fgfx_j$ where x_i and x_j are words of Mon. Using x_i and x_j we made explicit the context in which redexes may occur and we represented the way in which contexts are modified during a contraction step. Now we will derive the Diagram of Matchings for (a').
Since there is only one rule we will have only one node (a') .
Possible edges from (a') to (a') are due to the matchings of $x_i'fgfx_j'$ with $x_i ffx_j$. There are 4 possible matchings:

σ_1: $x_i' = x_i f$; $gfx_j' = x_j$ $\qquad$ σ_2: $x_i'fg = x_i$; $x_j' = fx_j$

σ_3: $x_i' = x_i ffa$; $x_j = bfgfx_j'$; $a = \ell sw(x_j)$; $b = rsw(x_i')$

σ_4: $x_i = x_i'fgfa$; $x_j' = bffx_j$; $a = \ell sw(x_j')$; $b = rsw(x_i)$

where ℓsw (or rsw) extracts a possibly empty left (or right) subword. Therefore the Diagram of Matchings is:



No infinite cycling is possible, because i) in cases σ_1 and σ_3 at each cycle at least one extra f in x_i' is used, and ii) in cases σ_2 and σ_4 at least one extra f in x_j' is used.

Example 12. [Der 79a]. In the Dutch National Flag problem we are given the following rewriting system:

a. $w,r \to r,w$

b. $b,r \to r,b$

c. $b,w \to w,b$

We will transform it by explicity introducing contexts, suppressing commas and introducing "]" (or "[") to the right (or left) of each r (or b). The presence of] and [shows that r can only terminate a redex and b can only begin it. We obtain:

a'. $x_i\ w\ r]\ x_j \to x_i\ r]\ w\ x_j$

b'. $x_i\,[b\ r]\ x_j \to x_i\ r]\,[b\ x_j$

c'. $x_i\,[b\ w\ \ x_j \to x_i\ w\ [b\ x_j$

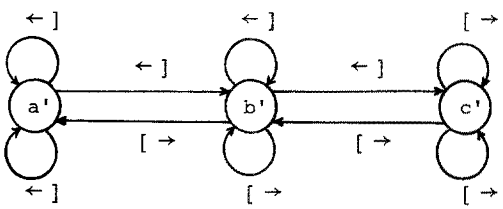
where x_i and x_j denote the contexts. They are words over the free monoid with symbols in {r,w,b,[,]}. Since redexes are made out of 3 symbols and the rules do not change the contexts, for studying the generation of new redexes, we need only to consider matchings generated by shifts up to 2 symbols. For the r.h.s. of the rule a', i.e. $x_i'\ r]\ w\ x_j'$, we have the following matchings:

a'-a': $x_i = x_i'\ r]$; $x_j' = r]\ x_j$ (] moves to the left: it will be denoted by $\leftarrow$])

a'-a': $x_i\ w = x_i'$; $x_j = w\ x_j'$ (again] moves to the left)

a'-b': $x_i\ [b = x_i'$; $x_j = w\ x_j'$ (again] moves to the left)

The complete Diagram of Matchings is:



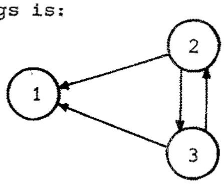
where $a' \xrightarrow{\leftarrow]} b'$ means that, after reducing an a'-redex, we can reduce an adjacent b'-redex only at the expenses of moving a right parenthesis to the left. Any infinite path in the Diagram Matchings determines an infinite movement of right parentheses to the left (or left parentheses to the right). Such a movement is impossible and therefore strong termination is proved.

<u>Example 13.</u> [Der 79c].

1. $\neg\neg x \to x$
2. $\neg(x \vee y) \to \neg\neg\neg x \wedge \neg\neg\neg y$
3. $\neg(x \wedge y) \to \neg\neg\neg x \vee \neg\neg\neg y$ where $F = \{\wedge, \vee, \neg, A\}$

$\wedge$ and $\vee$ have arity 2, $\neg$ has arity 1 and A has arity 0.

The Diagram of Matchings is:



where the edges to 1 stand for many edges, and $2 \longrightarrow 3$ is due to the fact that, given a 2-redex r, $\neg$r produces after its contraction, a 3-redex. Analogously for $3 \longrightarrow 2$. In order to prove termination we need to show that the unfolding of the strongly connected component $2 \rightleftarrows 3$ of the Diagram of Matching is finite, i.e. we need to show strong termination for the rules 2 and 3 only. Given $t \in T(F)$, if t has n binary operators, we consider the set of indexes $I = \{1,2,\ldots n\}$. We associate to each $\wedge$ and $\vee$ occurring in t a distinct index in I, so that an operator which is above another has a smaller index. During the reduction process those indexes are inherited in the obvious way (because one $\wedge$ produces only one $\vee$ and viceversa). We can associate with each $\wedge$ (or $\vee$) a c-number which counts the number of $\neg$'s above it. If we consider the sequence $\langle c_1,\ldots,c_n\rangle$ where c_i is the c-number of the operator with index i, then at each contraction that sequence decreases in lexicographic order. Therefore by Theorem 10 termination is guaranteed.

7. Conclusions

We proved that the NA property is a sufficient condition for strong termination of Combinatory Logic rewriting systems of proper combinators using the recursive path ordering (r.p.o.) method. This in particular solves the conjecture that all applicative combinations of the combinator +, s.t. + abcd $\rightarrow$ ac(bcd), have strong normal form [Böh 77]. We then compared various methods for proving strong termination and we proved that: i) r.p.o. with multiset orderings are in general not less powerful than r.p.o. with bounded lexicographic ordering as defined in [KaL 80], ii) for Weak Combinatory Logic with proper combinators, r.p.o. with bounded lexicographic ordering is equivalent to NA property, iii) if we represent the application of two terms t1 and t2 as the binary tree

```
   a
  / \
t1   t2
```

, we showed that NA property and r.p.o. with multiset ordering have incomparable power. Finally we presented some ideas for dealing with rewriting systems with nonproper combinators and we introduced a new technique for proving strong termination. It is called Diagram of Matchings and it can be useful for combining the strength of various different methods of proof.

8. Acknowledgements

Many thanks to J.J.Lévy at INRIA (Paris), and M.Venturini Zilli at IAC-CNR(Rome) who, through many stimulating conversations, gave me the enthusiasm to study the NA property. I am also grateful to all my friends at the Computer Science Department of Edinburgh University for their wonderful hospitality. In particular I wish to thank M.Smyth: a great deal is due to him and to his patience in talking to me. Thanks go to the CNR-Italy for the financial support and to Eleanor Kerse for typing a lengthy first draft of this paper.

8. References

[Böh 77] Böhm,C. Private communication: "All applicative combinations of + have strong normal form" (1977).

[Cur 58] Curry,H.B. and R.Feys: Combinatory Logic Vol. I. North-Holland Publishing Company, Amsterdam (1958).

[Der 79a] Dershowitz,N. and Z.Manna: "Proving termination with multiset orderings" C.A.C.M. Vol. 22 N. 8, 465-476.

[Der 79b] Dershowitz,N.: "A note on simplification orderings" Info. Proc. Lett. Vol. 9, N. 5, 212-215.

[Der 79c] Dershowitz,N.: "Orderings for Term Rewriting Systems" 20th Foundations of Computer Science Symposium (1979).

[Hue 77] Huet,G.: "Confluent Reductions: Abstract Properties and Applications to Term Rewriting Systems", 18th IEEE Symposium on Foundations of Computer Science (1977) 30-45.

[HuO 80] Huet,G. and D.Oppen: "Equations and rewrite rules: a survey", Technical Report CSL-111 (Jan. 80) SRI International, Menlo Park, California, U.S.A.

[KaL 80] Kamin,S. and J.J.Lévy: "Attempts for generalizing the Recursive Path Orderings" (unpublished manuscript 1980).

[Lév 80] Lévy,J.J. Personal communication (1980).

[Pet 75] Pettorossi,A.: "On termination of subrecursive classes of algorithms" (in Italian), AICA Congress 1975, Genova (Italy).

[Pet 78] Pettorossi,A.: "A Property which guarantees termination in Weak Combinatory Logic and Subtree Replacement Systems". Rapporto Istituto di Automatica, University of Rome (Italy) (1978).

[Pla 78] Plaisted,D.: "A recursively defined ordering for proving termination of term rewriting systems", Report R-78-943, Dept. of Computer Science, University of Illinois, Urbana, Ill. (1978).