11 Computations in Orthogonal Rewriting Systems, I

Gérard Huet and Jean-Jacques Lévy

We present in this chapter a study of derivations which formalize the computations of expressions in programming languages. Our formalism extends recursive programs with simplifications in [16, 20, 6, 27, 23, 25, 8, 2]. It permits to provide an operational semantics to programming languages such as LISP [19], LUCID [1], HOPE [5], and ML [9] The derivations we study may also be considered as proofs in certain equational theories which are of use to study abstract data types [11, 22, 6]. Finally, our work may be used for the design of efficient simplifications in formula manipulation systems such as LCF [10].

The results of this paper may be sketched as follows: We consider finite systems of pairs of first-order terms: $\Sigma = \{\alpha_i \to \beta_i | i \leq n\}$. Computations consist of rewriting an occurrence of some instance of an α_i into the corresponding instance of β_i. We restrict every rule $\alpha_i \to \beta_i$ so that all variables of β_i occur in α_i and so that every variable of α_i has a unique occurrence in α_i. Furthermore, no two rules of Σ are nontrivially superposable. We believe that most computations in programming languages without parallelism can be formalized in this setting.

Section 1 contains the preliminary definitions. In section 2 we show how the parallel moves lemma induces a congruence ("permutation") on derivations issued from a given term. The usual notion of residuals of a redex is extended to derivations, yielding a sup-semilattice structure to the permutation classes of derivations. This generalizes Church-Rosser results from [26, 13, 24].

In section 3 we prove what is usually called the standardization theorem. The problem is to simulate any derivation by a "standard" derivation, i.e., one which computes in an outside-in way For this we define the notion of a redex occurrence being external for a given derivation. Intuitively, an external redex is such that none of its residuals will be below any redex contracted in any permutation of the given derivation. This notion is thus invariant by permutation, and every nonempty derivation possesses at least one external redex occurrence. This allows us to define the notion of a standard derivation, and we show that every derivation class contains a unique standard derivation. We finally define the notion of a redex occurrence being external in a given term, which yields the notion of the normal derivation issued from a given term. When the term possesses a normal form, the normal derivation terminates in this normal form. This may be considered as an extension to our systems of the call by name computation rule for recursive program schemas, and of the normal derivations in λ-calculus. However, our formalism differs in essential ways from these two, because of the possibility of upward creation of redexes, and our call by name rule does not correspond to a simple lefmost-outermost strategy of computation. The results imply that all the nonambiguous linear term rewriting systems are d-outer in the sense of O'Donnell [24] The existence of normal derivations entails the possibility of doing only needed computations in a top-down manner. However, it does not correspond to an effective computation rule, and at this stage the only correct effective interpreters use parallel strategies such as parallel outermost.

Part II of this paper (chapter 12) will deal with the problem of finding effective interpreters which compute in a sequential fashion.

Most of our concepts are adaptations to syntactic domains of denotational semantics notions. This algebraic approach permits us to state and prove our theorems in an abstract fashion, suggesting that our results can be extended to a more general theory of operational semantics.

1 Preliminary Definitions

We follow here the notations of Huet [13] and Berry-Lévy [2]. Let $\mathscr{F}_n$ be a set of function symbols of arity n, $\mathscr{F} = \bigcup\{\mathscr{F}_n | n \geq 0\}$, and $\mathscr{V}$ a denumerable set of variable symbols. Our expression language is the set $\mathscr{M}(\mathscr{F},\mathscr{V})$ of first-order *terms* formed from $\mathscr{F}$ and $\mathscr{V}$, i.e.,

$$\mathscr{V} \subseteq \mathscr{M}(\mathscr{F},\mathscr{V}),$$

$$F \in \mathscr{F}_n \;\&\; \forall i \leq n\; M_i \in \mathscr{M}(\mathscr{F},\mathscr{V}) \Rightarrow F(M_1, M_2, \ldots, M_n) \in \mathscr{M}(\mathscr{F},\mathscr{V}).$$

When $\mathscr{F}$ and $\mathscr{V}$ are fixed from the context, we shall usually denote $\mathscr{M}(\mathscr{F},\mathscr{V})$ by $\mathscr{T}$.

For any term M we define its set of *occurrences* $\mathscr{O}(M)$ as a finite subset of the set N_+^* of finite sequences of positive integers as follows:

$\Lambda \in \mathscr{O}(M)$ (the empty occurrence)

$u \in \mathscr{O}(M_i) \Rightarrow iu \in \mathscr{O}(F(M_1, M_2, \ldots, M_n))$ for $1 \leq i \leq n$

Intuitively, an occurrence of M names a subterm of M by its access path. If $u \in \mathscr{O}(M)$, we define the *subterm of M at u* as the term M/u, defined inductively by

$$M/\Lambda = M,$$

$$F(M_1, M_2, \ldots, M_n)/iu = M_i/u.$$

Finally, if $u \in \mathscr{O}(M)$, we define for every term N the *replacement in M at u by N* as the term $M[u \leftarrow N]$, defined by

$$M[\Lambda \leftarrow N] = M,$$

$$F(M_1, M_2, \ldots, M_n)[iu \leftarrow N] = F(M_1, M_2, \ldots, M_i[u \leftarrow N], \ldots, M_n).$$

We shall also use the notation $\bar{\mathscr{O}}(M) = \{u \in \mathscr{O}(M) | M/u \notin \mathscr{V}\}$.

Example Let $M = F(G(x), A)$. We have $\mathscr{O}(M) = \{\Lambda, 1, 1.1, 2\}$ and $\bar{\mathscr{O}}(M) = \{\Lambda, 1, 2\}$, $M/1 = G(x)$, and $M[1 \leftarrow H(B)] = F(H(B), A)$.

The set of occurrences $\mathscr{O}(M)$ is partially ordered by the prefix ordering $u \preceq v$ iff $\exists w$ $uw = v$. In this case we shall define v/u as w. If $u \npreceq v$ and $v \npreceq u$, we say that u and v are *disjoint*, and write $u|v$. Finally, $u \prec v$ iff $u \preceq v$ and $u \neq v$.

A *substitution* is any function σ from $\mathscr{T}$ to $\mathscr{T}$ satisfying

$$\sigma(F(M_1, M_2, \ldots, M_n)) = F(\sigma(M_1), \sigma(M_2), \ldots, \sigma(M_n)).$$

In other words, σ is a morphism in the algebra of terms and is therefore uniquely determined from its value on variables.

We call *term rewriting system* (abbreviated TRS) any set Σ of pairs of terms $\alpha_i \to \beta_i$ such that $\mathscr{V}(\beta_i) \subseteq \mathscr{V}(\alpha_i)$, where $\mathscr{V}(M)$ is the set of variables appearing in M. We denote by red_Σ the set of left-hand sides α_i of Σ, which we call *redex schemes*. For any substitution σ and $\alpha \in \mathrm{red}_\Sigma$, $\sigma(\alpha)$ is called a *redex* of Σ. We denote by $\mathscr{R}_\Sigma(M)$ the set of redex occurrences in the term M. A term M such that $\mathscr{R}_\Sigma(M) = \varnothing$ is called a Σ-*normal form*. We denote by $\mathscr{NF}_\Sigma$ the set of terms in Σ-normal form. From now on, we shall assume that TRS Σ is fixed and therefore drop the subscript Σ except when needed.

We say that the term M *reduces to N at occurrence u using rule* $\alpha_k \to \beta_k$ iff there exists a substitution σ such that $M/u = \sigma(\alpha_k)$ and $N = M[u \leftarrow \sigma(\beta_k)]$. The pair $A = \langle M, u \rangle$ is called an *elementary derivation*, and we shall write A: $M \xrightarrow{u}_k N$. In this notation, k, u or A may be omitted. Note that M, u and k determine unambiguously $\sigma(x)$ for every x in $\mathscr{V}(\alpha_k)$, and therefore $\sigma(\beta_k)$ and N.

A *derivation* is a sequence $A = A_1 A_2 \ldots A_n$ of elementary derivations A_i: $M_i \to M_{i+1}$. We use AB for the concatenation of A and B, 0 for the empty derivation, and $|A|$ for the length of derivation A. We shall use the notation $A: M \xrightarrow{*} N$ to indicate that derivation A starts in M and ends in N.

Let $U = \{u_1, u_2, \ldots, u_n\}$ be a set of mutually disjoint redex occurrences in term M, and let $M/u_i = \sigma_i(\alpha_i)$. We call *elementary multiderivation* the pair $A = \langle M, U \rangle$ and write A: $M \xrightarrow{U} N$, where

$$N = M[u_i \leftarrow \sigma_i(\beta_i) | 1 \le i \le n] = M[u_1 \leftarrow \sigma_1(\beta_1)] \ldots [u_n \leftarrow \sigma_n(\beta_n)]$$

(Of course, the order of the u_i's is irrelevant). We say that A *contracts* the set U.

We define *multiderivations* in the same way and use the same notation as for derivations. Furtermore, if we want to emphasize the system Σ we use, we can write $M \xrightarrow{*}_\Sigma N$. We denote $\mathscr{D}(M)$ the set of multiderivations issued from M, and $\mathscr{F}(A)$ the final term reached by the (multi)derivation A. We say that A and B are *coinitial* if A, $B \in \mathscr{D}(M)$, *cofinal* if $\mathscr{F}(A) = \mathscr{F}(B)$. The notation AB, for the concatenation of A and B, assumes that $B \in \mathscr{D}(\mathscr{F}(A))$. Concatenation being associative, we write ABC for $(AB)C$ or $A(BC)$. Every set $\mathscr{D}(M)$ contains an empty multiderivation which we shall denote 0, the term M being usually understood from the context. Thus we freely write $A0 = 0A = A$.

When considering a derivation A: $M_0 \xrightarrow{u_1} M_1 \xrightarrow{u_2} M_2 \xrightarrow{u_3} \cdots \xrightarrow{u_n} M_n$, it is convenient to denote by $A[i]$ the i first steps of A, with $0 \le i \le n$. The rest of A is denoted $A[i, n]$. Thus $A = A[i]A[i, n]$, with M_i as the final term of $A[i]$. Similarly for multiderivations.

We shall study in this paper the properties of derivations in TRSs which have two constraints:

1) Left linearity: for every α in red, every variable of α occurs only once:

$\alpha/u = \alpha/v \in \mathscr{V} \Rightarrow u = v$

2) Nonambiguity: if α_i, $\alpha_j \in$ red, for every u in $\bar{\mathcal{O}}(\alpha_i)$ there are no σ, σ' such that $\sigma(\alpha_i/u) = \sigma'(\alpha_j)$, except in the trivial case $i = j$ and $u = \Lambda$

We also rule out the trivial TRSs which only consist of rules $\alpha \to \beta$ with $\alpha \in \mathscr{V}$ Therefore, we have red $\cap \mathscr{V} = \varnothing$ by condition (2)

DEFINITION 1.1 We call *orthogonal* any non-trivial TRS verifying conditions 1 and 2 above.

Our TRSs are similar to the schematic tree replacement systems of Rosen [26] and O'Donnell [24]. Condition 2 is called the nonoverlapping condition in these papers. Note that 1 and 2 imply together that our systems are outer, in the terminology of [24]. In the terminology of Knuth-Bendix [17], there is no critical pair. The derivation relation $\to$ is confluent (has the Church-Rosser property) [13, 26] We shall study in this paper some stronger properties of derivation spaces, generalizing results in Berry-Lévy [2] obtained for recursive equations.

2 The Derivations Space

In this section we prove an important property of derivations in TRSs, the parallel moves lemma (see Curry & Feys [7]). This allows us to define a partial ordering on derivations, inducing a semi-lattice property on derivation spaces. This useful tool generalizes the one defined in Berry-Lévy [2], which can also be defined in the λ-calculus (Lévy [18])

DEFINITION 2.1 Given an elementary derivation A: $M \xrightarrow{u}_k N$ and $v \in \mathscr{R}(M)$, we define the set $v\backslash A$ of *residuals* of v by A as a subset of $\mathcal{O}(N)$ as follows:

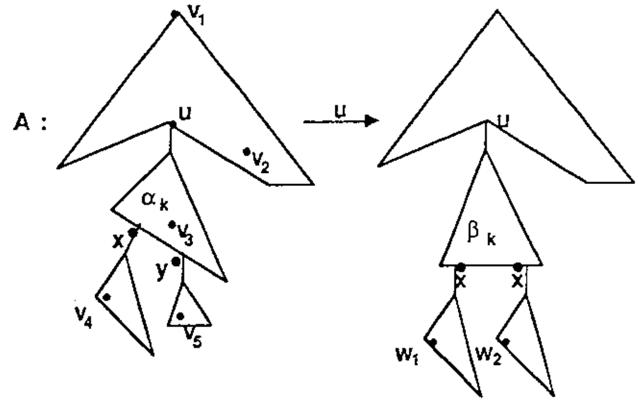
$$v\backslash A = \begin{cases} \varnothing & \text{if } v = u, \\ \{v\} & \text{if } v|u \text{ or } v \prec u, \\ \{uw_1v_1 \mid \beta_k/w_1 = x\} & \text{if } v = uwv_1 \text{ and } \alpha_k/w = x \in \mathscr{V}. \end{cases}$$

A pictorial explanation of residuals is given in figure 1. Note that all the relative positions of u and v have been considered because of nonambiguity. Furthermore, nonambiguity and left linearity imply that $v\backslash A \subseteq \mathscr{R}(N)$. Finally, for every w in $\mathscr{R}(N)$, there is at most one v such that $w \in v\backslash A$ When there is none, we say that w is *created* by A. Note that such ws may be above u, as well as in the substituted right-hand side β_k.

Example Let $\Sigma = \{F(G(x)) \to H(x, K(x)), F(H(x, y)) \to x, K(A) \to B, C \to A\}$

We consider

$$A: M_0 = F(F(H(G(C), C))) \xrightarrow{\{1\}} M_1 = F(G(C)) \xrightarrow{\{\Lambda\}} M_2 = H(C, K(C)) \xrightarrow{\{1,2.1\}} M_3 = H(A, K(A)) \xrightarrow{\{2\}} H(A, B).$$



$v_1\backslash A = \{v_1\}, v_2\backslash A = \{v_2\}, v_3\backslash A = v_5\backslash A = u\backslash A = \emptyset, v_4\backslash A = \{w_1, w_2\}$.

Figure 1

The redex occurrence 1.1.1.1 of M_0 has one residual in M_1, namely 1.1, and two residuals in M_2, namely 1 and 2.1. It has no further residuals after the second step of A, which reduces these two redexes. The redex occurrence 1.1.2 of M_0 has no residual in M_1. The redex occurrence 2 of M_3 is created (upward) by redex occurrence 2.1 of M_2. Finally, note that the redex occurrence Λ of M_2 is created by the first step of A, even though the replacing term does not contribute to its redex, since it is a variable!

For any nonelementary derivation A, we define $v\backslash A$ by

$$v\backslash 0 = \{v\},$$

$$v\backslash(AB) = \{w\backslash B \mid w \in v\backslash A\}$$

The residual mapping is extended to sets of redex occurrences by defining

$$U\backslash A = \bigcup \{u\backslash A \mid u \in U\}.$$

Finally, if A is an elementary multiderivation A: $M \xrightarrow{U} N$ with $U = \{u_1, u_2, \ldots, u_n\}$, we define $v\backslash A$ as $v\backslash(A_1 A_2 \ldots A_n)$, where A_i: $M_{i-1} \xrightarrow{u_i} M_i$, $M_0 = M$ and $M_n = N$. Of course, $v\backslash A$ is independent of the order of the u_i's. We extend $v\backslash A$ to nonelementary multiderivations and to sets of redex occurrences as above.

Let $A, B \in \mathscr{D}(M)$, with $|B| = 1$, contracting the set $U \subseteq \mathscr{R}(M)$. We define the *residual* $B\backslash A$ of B by A as the elementary derivation in $\mathscr{D}(\mathscr{F}(A))$, contracting the set $U\backslash A$. We also define $A \sqcup B = A(B\backslash A)$.

LEMMA 2.2: THE PARALLEL MOVES LEMMA Let $A, B \in \mathscr{D}(M)$, with $|A| = |B| = 1$. Then $\mathscr{F}(A \sqcup B) = \mathscr{F}(B \sqcup A)$, and for all u in $\mathscr{R}(M)$ we have $u\backslash(A \sqcup B) = u\backslash(B \sqcup A)$.

Proof Similar to the proof of theorem 6.5 in [23], lemma 11 in [17], lemma 2.1.6 in [3]. ■

The parallel moves lemma is illustrated in figure 2.

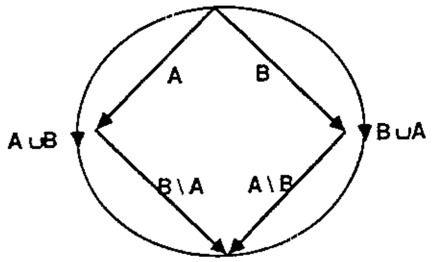


Figure 2

COROLLARY: THE CHURCH-ROSSER PROPERTY Since $|B\backslash A| = |A\backslash B| = 1$, lemma 2.2 shows that the multiderivation relation is strongly confluent (see [13]) and therefore that the relation $\to$ is confluent. Therefore, for every M, there is at most one normal form N such that $M \xrightarrow{*} N$.

The parallel moves lemma permits us to generalize the residual relation to arbitrary derivations, as follows.

DEFINITION 2.3 Let $A, B \in \mathscr{D}(M)$, with $|B| = 1$. We define $A\backslash B \in \mathscr{D}(\mathscr{F}(B))$ by induction on $|A|$:

$$0\backslash B = 0 \tag{1}$$

$$(A_1 A_2)\backslash B = (A_1\backslash B)(A_2\backslash(B\backslash A_1)) \quad \text{with } |A_1| = 1. \tag{2}$$

Note that $A_2\backslash(B\backslash A_1)$ is defined by induction, since $|B\backslash A_1| = 1$, and that it can be concatenated to $A_1\backslash B$, by the preceding lemma. Now for A, $B \in \mathscr{D}(M)$ of arbitrary lengths, we define $A\backslash B \in \mathscr{D}(\mathscr{F}(B))$ by induction on $|B|$:

$$A\backslash 0 = A \tag{3}$$

$$A\backslash(B_1 B_2) = (A\backslash B_1)\backslash B_2 \quad \text{with } |B_1| = 1 \tag{4}$$

We also extend the notation $A \sqcup B = A(B\backslash A)$ to derivations of any length. It is easy to show by induction that $|A\backslash B| = |A|$, that equations (1) through (4) are valid without length conditions, and to generalize the parallel moves lemma:

LEMMA 2.4: THE GENERALIZED PARALLEL MOVES LEMMA For all A, $B \in \mathscr{D}(M)$ we have $\mathscr{F}(A \sqcup B) = \mathscr{F}(\mathscr{B} \sqcup A)$ and for all u in $\mathscr{R}(M)$, $u\backslash(A \sqcup B) = u\backslash(B \sqcup A)$.

The last part of lemma 2.4 can be generalized to an arbitrary multiderivation coinitial with A and B as follows.

LEMMA 2.5: THE CUBE LEMMA For all $A, B, C \in \mathscr{D}(M)$ we have $C\backslash(A \sqcup B) = C\backslash(B \sqcup A)$.

Proof By induction on $|A| + |B| + |C|$. If $C = 0$, use (1); otherwise, let $C = C_1 C_2$ with $|C_2| = 1$. We get

$$\begin{aligned}(A \sqcup B)\backslash C_1 &= (A\backslash C_1)((B\backslash A)\backslash(C_1\backslash A)) && \text{by (2)}\\ &= (A\backslash C_1)(B\backslash A \sqcup C_1)) && \text{by (4)}\end{aligned}$$

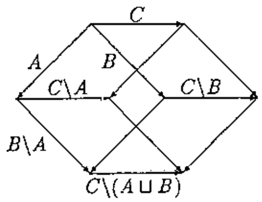


Figure 3

$$
\begin{aligned}
&= (A\backslash C_1)(B\backslash(C_1 \sqcup A)) && \text{by ind. hyp.}\\
&= (A\backslash C_1)((B\backslash C_1)\backslash(A\backslash C_1)) && \text{by (4)}\\
&= (A\backslash C_1)\sqcup(B\backslash C_1).
\end{aligned}
$$

Thus

$$
\begin{aligned}
C_2\backslash((A\sqcup B)\backslash C_1) &= C_2\backslash((A\backslash C_1)\sqcup(B\backslash C_1))\\
&= C_2\backslash((B\backslash C_1)\sqcup(A\backslash C_1)) && \text{by lemma 2.4}\\
&= C_2\backslash((B\sqcup A)\backslash C_1). && \text{symmetrically}
\end{aligned}
$$

Finally, $C_1\backslash(A\sqcup B) = C_1\backslash(B\sqcup A)$ by the induction hypothesis, which achieves the proof, using (2). ■

The cube lemma is illustrated in figure 3.

If we now define, for $A, B \in \mathscr{D}(M)$, $A \equiv B$ iff for all $C \in \mathscr{D}(M)$, $C\backslash A = C\backslash B$, we get the following corollary:

COROLLARY $\forall A, B \in \mathscr{D}(M), A \sqcup B \equiv B \sqcup A$

We shall call $\equiv$ *permutation* equivalence.

LEMMA 2.6: $\sqcup$ IS ASSOCIATIVE $\forall A, B, C \in \mathscr{D}(M), (A\sqcup B)\sqcup C = A\sqcup(B\sqcup C)$

Proof

$$
\begin{aligned}
(A\sqcup B)\sqcup C &= (A\sqcup B)(C\backslash(A\sqcup B))\\
&= (A\sqcup B)(C\backslash(B\sqcup A)) && \text{by lemma 2.5}\\
&= A(B\backslash A)((C\backslash B)\backslash(A\backslash B)) && \text{by (4)}\\
&= A((B\sqcup C)\backslash A) && \text{by (2)}\\
&= A\sqcup(B\sqcup C)
\end{aligned}
$$
■

The empty multiderivation 0 in $\mathscr{D}(M)$ should not be confused with the elementary multiderivation starting from M and contracting an empty set of redex occurrences, which we shall denote $\varnothing$. In particular, $|\varnothing| = 1$. However, it follows from the definition of residual that for every A in $\mathscr{D}(M)$ we have $\varnothing\backslash A = \varnothing$. By an easy induction on $|A|$ we get that $A\backslash\varnothing = A$, and thus $0 \equiv \varnothing$, and also that $A\backslash A = \varnothing^n$, with $n = |A|$. This and lemma 2.4 easily imply that if $A \equiv B$, then $\mathscr{F}(A) = \mathscr{F}(B)$.

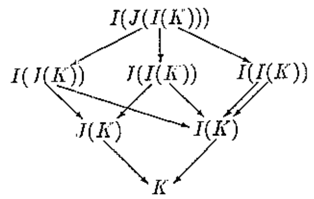


Figure 4

LEMMA 2.7: $\equiv$ IS A CONGRUENCE Let $A, B \in \mathscr{D}(M)$, with $A \equiv B$. We have

$\forall C \in \mathscr{D}(M)$, $A \backslash C \equiv B \backslash C$

$\forall C \in \mathscr{D}(M)$, $A \sqcup C \equiv B \sqcup C$

$\forall C \in \mathscr{D}(\mathscr{F}(A))$, $AC \equiv BC$

$\forall C$ such that $\mathscr{F}(C) = M$, $CA \equiv CB$.

Proof Easy consequences of the definitions. ■

COROLLARIES $\forall A \in \mathscr{D}(M)$, $A \backslash A \equiv 0$. $\forall A \in \mathscr{D}(M)$, $A \sqcup A \equiv A$.

We now define a relation $\sqsubseteq$ between coinitial multiderivations as follows:

$\forall A, B \in \mathscr{D}(M)$, $A \sqsubseteq B$ iff $A \sqcup B \equiv B$

THEOREM 2.8: LATTICE OF DERIVATIONS THEOREM $\langle \mathscr{D}(M)/\equiv, \sqsubseteq, \sqcup \rangle$ is an upper semi-lattice.

Proof Easy algebraic manipulations from the preceding lemmas. ■

Phrased in categorical terms, this result means that the category whose objects are terms and whose morphisms are the equivalence classes of derivations admits pushout

Caution! The lattice structure given by the parallel moves theorem is on *derivations* and *not* on terms. For instance, if we consider the system R consisting solely of the rules $I(x) \to x$ and $J(x) \to x$, figure 4 shows that the terms $I(J(K))$ and $J(I(K))$ do not possess a least upper bound. Note that this phenomenon may be traced to the existence of two non-equivalent derivations between $I(I(K))$ and $I(K)$. This shows that the categorical viewpoint is the right one here: we need to talk in terms of arrows, not just relations between terms. And now the confluence diagrams can be replaced by more informative commuting diagrams expressing permutation equivalences of derivations. For instance, in figure 4 certain sub-diagrams are confluent only for unimportant syntactic coincidences (due to the double occurrence of I in $I(J(I(K)))$), but the others are commuting diagrams expressing a strong equivalence of computations.

Using all that precedes, it is easy to prove the following relations (with the appropriate conditions on initial and final terms of the following derivations)

$$(AB) \backslash A \equiv B \tag{5}$$

$$(CA) \backslash (CB) \equiv A \backslash B \tag{6}$$

$$A \sqsubseteq B \text{ iff } A \backslash B = \varnothing^m, \text{ with } m = |A| \tag{7}$$

$$A \equiv B \text{ iff } A \sqsubseteq B \sqsubseteq A \tag{8}$$

$$A \equiv 0 \text{ iff } A = \varnothing^n \text{ for some } n \geq 0 \tag{9}$$

$$A \sqsubseteq B \text{ iff } AC \equiv B \text{ for some } C \tag{10}$$

$$A \sqsubseteq B \text{ implies } A \backslash C \sqsubseteq B \backslash C \tag{11}$$

$$A \sqsubseteq B \text{ implies } CA \sqsubseteq CB \tag{12}$$

Note that (7) and (8) show that $\sqsubseteq$ and $\equiv$ are easily decidable

We could also have defined $\equiv$ as the smallest congruence relation on $\mathscr{D}(M)$ such that $\varnothing \equiv 0$ and $A \sqcup B \equiv B \sqcup A$. This justifies our terminology of permutation equivalence

Finally, we may extend relations $\sqsubseteq$ and $\equiv$ to derivations by confusing the derivation

$$M_0 \xrightarrow{u_1} M_1 \xrightarrow{u_2} M_2 \xrightarrow{u_3} \cdots \xrightarrow{u_n} M_n$$

with the multiderivation

$$M_0 \xrightarrow{U_1} M_1 \xrightarrow{U_2} \cdots \xrightarrow{U_n} M_n,$$

where $\forall i \leq n$, $U_i = \{u_i\}$

3 Standardization, Call by Name, Call by Need

We are going to show in this section that every derivation is equivalent by permutations to a certain derivation computing redexes in an outside-in manner. In analogy to what happens for recursive definitions or λ-calculus, we shall call these special derivations *standard*. However, it should be remarked that, unlike in these two formalism, the leftmost outermost derivations are usually not standard in TRS. For instance, consider

$$\Sigma = \{F(x, A) \to B, C \to C, D \to A\}.$$

The standard derivation starting from the term $F(C, D)$ and ending in its normal form is

$$F(C, D) \to F(C, A) \to B,$$

whereas the leftmost outermost rule leads to the infinite derivation

$$F(C, D) \to F(C, D) \to \cdots$$

3.1 Outside-In and Standard Computations

Let us first give some preliminary definitions and technical lemmas.

DEFINITION 3.1 Let $u \in \mathscr{R}(M)$, α the redex scheme at u in M. We define the *contractum in M at u* as the set $C_M(u)$ of occurrences of M that are inside α:

$C_M(u) = \{uv \in \mathcal{O}(M) | v \in \bar{\mathcal{O}}(\alpha)\}$

We also define

$C_M(u) = \{uv \in \mathcal{O}(M) | v \notin \mathcal{O}(\alpha)\}$.

Note that $u \in C_M(u)$, since $\alpha \notin \mathscr{V}$, and thus $\forall v \in \overline{C_M}(u)$, $u \prec v$.

Let $u, v \in \mathscr{R}(M)$, with $u \prec v$. The nonambiguity condition imposes $v \in \overline{C_M}(u)$. Furthermore, for any elementary $A: M \xrightarrow{V} N$ such that $v \in V$, we get $u \backslash A = \{u\}$.

LEMMA 3.2 Let $u, v \in \mathscr{R}(M)$. If $C_M(u) \cap \overline{C_M}(v) \neq \varnothing$, then $u \in \overline{C_M}(v)$.

Proof Easy corollary of nonambiguity. ■

DEFINITION 3.3 Let $A: M_0 \xrightarrow{U_1} M_1 \xrightarrow{U_2} M_2 \xrightarrow{U_3} \cdots \xrightarrow{U_n} M_n$, and $u \in \mathcal{O}(M_0)$. We say that A *preserves* u iff A does not contract a redex above u, that is, $\forall i \leq n\ \nexists v \in U_i$, $v \prec u$.

LEMMA 3.4 Let $A \in \mathscr{D}(M)$, preserving v. For every u in $\mathscr{R}(M)$ such that $u \prec v$, we have $u \backslash A = \{u\}$.

Proof Easy induction on $|A|$. ■

We can easily refine the parallel moves lemma to derivations preserving an occurrence u, using:

LEMMA 3.5 Let A, B be coinitial, both preserving u. Then $A \backslash B$ preserves u.

Proof Easy induction on $|A| + |B|$, using the following observation. Let $A_1: P \xrightarrow{v} Q$. For all $w \in \mathscr{R}(P)$, $w \backslash A_1$ does not contain any occurrence above both v and w. ■

LEMMA 3.6 Let A, B be coinitial, with $A \sqsubseteq B$. If B preserves u, then A too preserves u.

Proof Assume that $A \sqsubseteq B$, B preserves u, and A does not preserve u. That is, $A = A_1 A_2 A_3$, where A_1 preserves u and A_2 contracts set W such that $\exists w \in W$. By lemma 3.5, $B \backslash A_1$ preserves u, and by lemma 3.4, $w \backslash (B \backslash A_1) = \{w\}$, whence $A \backslash B \not\equiv 0$, contrary to hypothesis. ■

We are now ready to present the important definition of an occurrence being external for a (multi)derivation.

DEFINITION 3.7 Let $A \in \mathscr{D}(M)$, u in $\mathcal{O}(M)$. We say that u is *external* for A, and write $u \in \mathscr{X}(A)$, iff either A preserves u or $A = A_1 A_2 A_3$, and there exists $v \prec u$ such that A_1 preserves u, $A_2: P \xrightarrow{V} Q$, with $v \in V$ and $u \in C_P(v)$ and $v \in \mathscr{X}(A_3)$.

Note that in the second case v and the decomposition of A into $A_1 A_2 A_3$ are unique: A_2 is the first step in A that does not preserve u, and v is the unique redex occurrence above u contracted at this step. Intuitively, u is external for A iff A does not contract at some step a redex above u for which the symbol at u in M did not contribute.

Example Let $\Sigma = \{F(x, B) \to G(x, x), A \to B\}$ and consider

$A: M_0 = F(C, A) \to M_1 = F(C, B) \to M_3 = G(C, C)$

We have $\mathscr{X}(A) = \{\Lambda, 2\}$

DEFINITION 3.8 Let $A: M_0 \xrightarrow{U_1} M_1 \xrightarrow{U_2} M_2 \xrightarrow{U_3} \cdots \xrightarrow{U_n} M_n$. We say that u is an *initial redex occurrence contributing* to A, and write $u \in \mathscr{R}(A)$, iff $\exists i \leq n\ U_i \cap (u \backslash A[i-1]) \neq \varnothing$. Finally, we define the *external redex occurrences* of A as $\mathscr{E}(A) = \mathscr{R}(A) \cap \mathscr{X}(A)$.

Note that outermost redex occurrences in the starting term of A may be excluded from $\mathscr{E}(A)$ either because they are not external for A or else because they are not contracted in A

We shall first extend lemmas 3.4, 3.5 and 3.6 above to $\mathscr{X}(A)$. For lemma 3.4 we need an extra hypothesis: v must be below the redex scheme at u.

LEMMA 3.9 Let $A \in \mathscr{D}(M)$, u in $\mathscr{R}(M)$. If $\mathscr{X}(A) \cap \overline{C_M}$, we have $u \backslash A = \{u\}$.

Proof By induction on $|A|$ Let $v \in \mathscr{X}(A) \cap \overline{C_M}(u) \neq \varnothing$. If A preserves v, use lemma 3.4. Otherwise, $A = A_1 A_2 A_3$, and there exists $w \prec v$ such that the following hold:

a A_1 preserves v, and therefore $u \backslash A_1 = \{u\}$ by lemma 3.4

b. $A_2: P \xrightarrow{W} Q$ with $w \in W$ and $v \in C_P(w)$ Furthermore, by (a), $v \in \overline{C_P}(u)$, and thus $w \in \overline{C_P}(u)$, by lemma 3.2 Therefore, $u \backslash A_2 = \{u\}$, and $w \in \overline{C_Q}(u)$.

c Also, $w \in \mathscr{X}(A_3)$ and therefore, using $w \in \overline{C_Q}(u)$, we get $u \backslash A_3 = \{u\}$, by the induction hypothesis, and finally, $u \backslash A = \{u\}$. ■

We now extend lemma 3.5 as follows.

LEMMA 3.10 Let A and B be coinitial and B preserve $u \in \mathscr{X}(A)$. Then $u \in \mathscr{X}(A \backslash B)$

Proof By induction on $|A|$.

Case 1: A preserves u. Then by lemma 3.5, $A \backslash B$ preserves u, and therefore $u \in (A \backslash B)$.

Case 2: $\exists v \prec u$ such that $A = A_1 A_2 A_3$, as in the definition above. As A_1 and B preserve u, $A_1 \backslash B$ preserves u, by lemma 3.5. Similarly, $B \backslash A_1$ preserves u, and therefore $v \backslash (B \backslash A_1) = \{v\}$, by lemma 3.4. Finally, $B \backslash A_1$ preserves v (since $v \prec u$), A_2 preserves v trivially, and by lemma 3.5, $B \backslash (A_1 A_2)$ preserves v. Now $v \in \mathscr{X}(A_3)$ implies $v \in \mathscr{X}(A_3 \backslash (B \backslash (A_1 A_2)))$, by the induction hypothesis, and therefore,

$$A \backslash B = (A_1 \backslash B)(A_2 \backslash (B \backslash A_1))(A_3 \backslash (B \backslash (A_1 A_2)))$$

is a decomposition, which shows that $u \in \mathscr{X}(A \backslash B)$ ■

Note also that it follows from the definition of $\mathscr{X}$ that if A preserves $u \in \mathscr{X}(B)$, we have $u \in \mathscr{X}(AB)$ We shall freely use this property below. We are now able to prove the extension of lemma 3.6

LEMMA 3.11 Let A and B be coinitial, with $A \sqsubseteq B$. We have $\mathscr{X}(B) \subseteq \mathscr{X}(A)$.

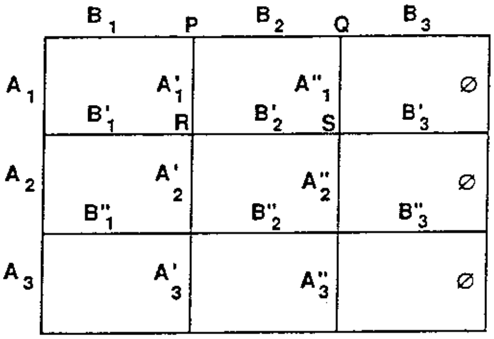


Figure 5

Proof Let $A \sqsubseteq B$ and $u \in \mathscr{X}(B)$. We show that $u \in \mathscr{X}(A)$ by induction on $|A|$.

Case 1: B preserves u. Then A preserves u, by lemma 3.6, and thus $u \in \mathscr{X}(A)$.

Case 2: Otherwise, $B = B_1 B_2 B_3$, with $B_2 \colon P \xrightarrow{V} Q$; $\exists v \in V$, $v \prec u$; B_1 preserves u; $u \in C_P(v)$; and $v \in \mathscr{X}(B_3)$. Since A does not preserve u, we have $A = A_1 A_2 A_3$, where A_1 preserves u and $A_2 \colon R \xrightarrow{W} S$ such that $\exists w \in W$ $w \prec u$. Let $A_1', A_2', \ldots, B_1', B_2', \ldots$ be as shown in figure 5, which expresses the parallel moves lemma for A and B. By lemma 3.5, A_1' preserves u and therefore v, B_2 preserves v trivially, and by lemma 3.5, again A_1'' preserves v. By lemma 3.10 we get $v \in \mathscr{X}(B_3')$. As A_1' preserves u, v is contracted by B_2', by lemma 3.4. Similarly, B_1' preserves u and therefore w, according to the relative positions of $v, w \in \mathscr{R}(R)$.

Case 2.1: $v \neq w$. Since our systems are nonambiguous, $u \in C_R(v)$ and $w \prec u$ imply $v \in \overline{C_R}(w)$, and thus $w \backslash B_2' = \{w\}$. Therefore, $w \in \mathscr{R}(S)$. Using $v \in \mathscr{X}(B_3')$, we get by lemma 3.9 that $w \backslash B_3' = \{w\}$, and therefore $A_2 \backslash (B_1' B_2' B_3') \neq \varnothing$, a contradiction with $A \sqsubseteq B$.

Case 2.2: $v = w$. Since A_2' and B_2' preserve v, so do A_2'' and B_2'', by lemma 3.5. By lemma 3.10 we get $v \in \mathscr{X}(B_3'')$. As $B_1'' B_2''$ preserve v, we have $v \in \mathscr{X}(B_1'' B_2'' B_3'')$. Since $A_3 \sqsupseteq B_1'' B_2'' B_3''$, we get $v \in \mathscr{X}(A_3)$, by the induction hypothesis, and that $A_1 A_2 A_3$ is a decomposition of A, which shows that $u \in \mathscr{X}(A)$. ■

COROLLARY $A \equiv B$ implies $\mathscr{X}(A) = \mathscr{X}(B)$.

External redex occurrences are their own residual until they are contracted:

LEMMA 3.12 Let $A \colon M \xrightarrow{*} N$ and $u \in \mathscr{X}(A) \sqcup \mathscr{R}(M)$. We have either $u \in \mathscr{R}(A)$ and $u \backslash A = \varnothing$ or $u \notin \mathscr{R}(A)$ and $u \backslash A = \{u\}$.

Proof Straightforward from the definitions of $\mathscr{X}(A)$ and $\mathscr{R}(A)$. ■

LEMMA 3.13 If $A \equiv B$, we have $\mathscr{E}(A) = \mathscr{E}(B)$.

Proof Let $A \equiv B$ and $u \in \mathscr{E}(A)$. We get $u \in \mathscr{X}(B)$ by the corollary to lemma 3.11.

Lemma 3.12 implies $u\backslash A = \varnothing$ and therefore, by the parallel moves lemma, $u\backslash B = \varnothing$. Now lemma 3.12 applied to B gives $u \in \mathscr{R}(B)$, and thus $u \in \mathscr{E}(B)$. ∎

If u is preserved by A, then any prefix occurrence $v \preceq u$ is also preserved by A, obviously. This property is also true of members of $\mathscr{X}(A)$.

LEMMA 3.14 If $u \in \mathscr{X}(A)$, then $\forall v \preceq u\ v \in \mathscr{X}(A)$.

Proof Induction on $|A|$.

Case 1: A preserves u. Then A preserves v, and $uv \in \mathscr{X}(A)$.

Case 2: $A = A_1 A_2 A_3$ with A_2: $P \xrightarrow{W} Q$ and $\exists w \prec u$ such that $w \in W$, $u \in C_P(w)$ and $w \in \mathscr{X}(A_3)$. Moreover, A_1 preserves u and therefore v.

Case 2.1: $u \preceq w$. Then $v \in \mathscr{X}(A_3)$ by the induction hypothesis, and therefore, since $A_1 A_2$ preserve v, we get $v \in \mathscr{X}(A)$.

Case 2.2: $w \prec v \preceq u$. Then $v \in C_P(w)$, and therefore the decomposition $A_1 A_2 A_3$ shows that $v \in \mathscr{X}(A)$.

The set $\mathscr{X}(A)$ is therefore closed by prefix. When it contains some $u \in \mathscr{R}(M)$, with $A \in \mathscr{O}(M)$, is also contains all the members of $C_M(u)$. $\mathscr{X}(A)$ is obviously non-empty, since it always contains Λ, preserved by every derivation. We shall now show that $\mathscr{E}(A)$ is also non-empty whenever $A \neq \varnothing^n$.

LEMMA 3.15 If $A \not\equiv 0$, we have $\mathscr{E}(A) \neq \varnothing$.

Proof By induction on $|A|$. If $A = 0$, the proof is trivial. Otherwise, let $A = A_1 A_2$, with $A_1 : M \xrightarrow{U} N$ elementary. If $A_2 \equiv 0$, then $U \neq \varnothing$, and any member of U is obviously in $\mathscr{E}(A)$. Otherwise, let us consider $v \in \mathscr{E}(A_2)$, which exists by the induction hypothesis.

Case 1: There exists u in U such that $u \preceq v$. Then $u \in \mathscr{X}(A_2)$ by lemma 3.14 and thus $u \in \mathscr{X}(A)$ since A_1 preserves u. Therefore, $u \in \mathscr{E}(A)$.

Case 2: Otherwise, v is preserved by A_1, and therefore $v \in \mathscr{X}(A)$.

Case 2.1: There exists u in $U \cap C_M(v)$. Then by definition, $u \in \mathscr{X}(A)$, and therefore $u \in \mathscr{E}(A)$.

Case 2.2: Otherwise, $v \in \mathscr{R}(A)$, and therefore $v \in \mathscr{E}(A)$. ∎

We are now ready to use $\mathscr{E}(A)$ to construct an outside-in equivalent of A. The following lemma is useful to show that this construction will always terminate.

LEMMA 3.16 There is no infinite chain $A_1 \rhd A_2 \rhd A_3 \rhd \cdots$, where $A \rhd B$ iff $A \equiv B$ and $B = A\backslash u$, with $u \in \mathscr{E}(A)$.

Proof Let A: $M_0 \xrightarrow{U_1} M_1 \xrightarrow{U_2} M_2 \xrightarrow{U_3} \cdots \xrightarrow{U_n} M_n$. If $A \equiv 0$, there is no B such that $A \rhd B$. Otherwise, let $u \in \mathscr{E}(A)$. According to lemma 3.12, there exists $k \leq n$ such that $u\backslash A[k-1] = \{u\}$ and $u \in U_k$. Therefore, $(A\backslash u)[k-1, n] = A[k-1, n]\backslash u$ contracts the sets $U_k - \{u\}, U_{k+1}, \ldots, U_n$. Thus $A\backslash u$ is less than A in the lexicographic ordering on the tuples $\langle |U_n|, |U_{n-1}|, \ldots, |U_1| \rangle$. ∎

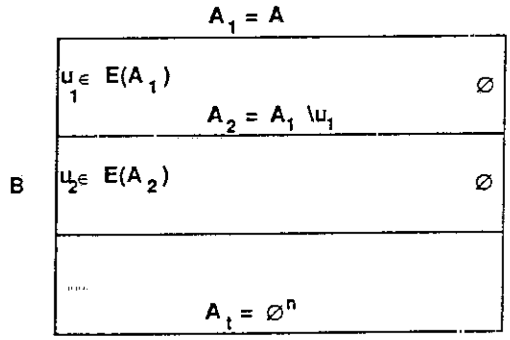


Figure 6

DEFINITION 3.17 The derivation A is said to be *outside-in* iff either $A = 0$ or $A = A_1 A_2$, where A_1 is the elementary derivation contracting some u in $\mathscr{E}(A)$ and A_2 is outside-in

In other words, in an outside-in derivation, we contract at every step some redex occurrence external for the rest of the derivation.

Let A be any multiderivation. We construct an outside-in equivalent B of A as follows If $A \equiv 0$, we stop with $B = 0$ Otherwise, let us pick some u in $\mathscr{E}(A)$. We define B as the elementary derivation contracting u, followed by an outside-in equivalent of $A \backslash u$. By Nœtherian induction, using lemma 3.16, the construction always terminates. By construction, $B \sqsubseteq A$. But according to lemma 3.15, the construction can only stop when $A \sqsubseteq B$, and therefore $B \equiv A$, which justifies our terminology. The construction is illustrated in figure 6.

DEFINITION 3.18 Let A be any derivation We say that A is *standard* iff A is outside-in and at every step the redex occurrence u is the *leftmost* in $\mathscr{E}(A)$

Similarly to the above, we define *the standard derivation B* in the class of A, and write $B = \mathrm{st}(A)$ Using lemma 3.13, we easily get by Nœtherian induction that $\mathrm{st}(A_1) = \mathrm{st}(A_2)$ whenever $A_1 \equiv A_2$.

To conclude, we have shown the following:

THEOREM 3.19: STANDARDIZATION THEOREM Every derivation class contains a unique standard derivation.

Note that the leftmost condition is unimportant here: any choice function over sets of disjoint occurrences would guarantee the uniqueness of the standard derivation

Example Let $\Sigma = \{F(x, B) \to G(x, x), A \to B, C \to D\}$. Among the derivations starting at $F(C, A)$, all the ones that do not use the reductions marked with an X in figure 7 are standard.

Finally, we remark that it is possible to express standard derivations in terms of residuals, because $\mathscr{E}(A)$ can be characterized as the set of redex occurrences of A that stay outermost modulo permutations in the following sense.

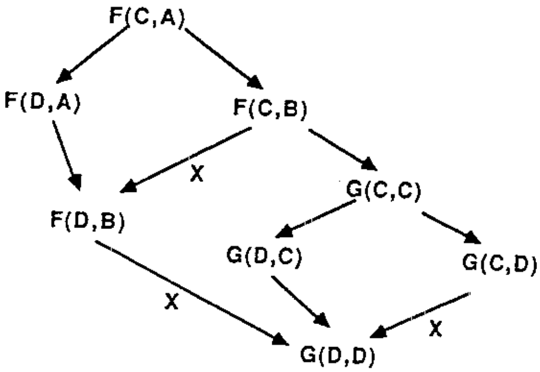


Figure 7

DEFINITION 3.20 Let $A: M_0 \xrightarrow{U_1} M_1 \xrightarrow{U_2} M_2 \xrightarrow{U_3} \cdots \xrightarrow{U_n} M_n$ and $u \in \mathcal{R}(A)$. We say that u is *outermost* in A, and write $u \in \mathrm{Out}(A)$, iff $\forall i \leq n\ \forall v \in u\backslash A[i-1]\ \nexists u_i \in U_i\ u_i \prec v$.

In other words, $u \in \mathrm{Out}(A)$ iff $u \in U$ for some $i \leq n$, and $\forall j \prec i$, $\nexists u_j \in U_j$, $u_j \preceq u$. Therefore, $\forall j \prec i, u\backslash A[j] = \{u\}$ and $u\backslash A[i] = u\backslash A = \varnothing$. Note that lemma 3.12 implies that $\mathscr{E}(A) \subseteq \mathrm{Out}(A)$. The converse is true for outside-in derivations.

LEMMA 3.21 If A is outside-in, we have $\mathscr{E}(A) = \mathrm{Out}(A)$

Proof Let $A: M_0 \xrightarrow{u_1} M_1 \xrightarrow{u_2} M_2 \xrightarrow{u_3} \cdots \xrightarrow{u_n} M_n$ be an outside-in derivation and $u \in \mathrm{Out}(A)$. By definition, $\exists i \leq n$, $u_i = u$ and $\forall j \prec i$, $u_j \not\preceq u$. Since A is outside-in, we have $u \in \mathscr{E}(A[i-1, n])$ and, since $A[i-1]$ preserves u, we get $u \in \mathscr{E}(A)$. The converse follows from lemma 3.12. ■

LEMMA 3.22 $\mathscr{E}(A) = \bigcap_{B \equiv A} \mathrm{Out}(B)$.

Proof Let $B \equiv A$. We have, by lemma 3.13, $\mathscr{E}(A) = \mathscr{E}(B) \subseteq \mathrm{Out}(B)$, which shows that $\mathscr{E}(A) \subseteq \bigcap_{B \equiv A} \mathrm{Out}(B)$. For the converse, take $B = \mathrm{st}(A)$ and use the preceding lemma. ■

That is, $\mathscr{E}(A)$ is the set of redex occurrences which are outermost in every permutation of A, or equivalently, which are outermost in some outside-in equivalent of A.

The results obtained so far do not allow us to say how to compute in a standard way, since $\mathscr{E}(A)$ may depend on the whole of A. We are now going to relativize these notions to the starting term of derivation A, so that we may define an outside-in computation rule, generalizing the call-by-name computation rule for recursive program schemes.

3.2 Normal Derivation, Call by Name

DEFINITION 3.23 We define the set of *external occurrences* in term M as $\mathscr{X}(M) = \bigcap_{A \in \mathscr{D}(M)} \mathscr{X}(A)$. Similarly to above, we define the set of *external redex occurrences* in M as $\mathscr{E}(M) = \mathscr{R}(M) \cap \mathscr{X}(M)$, or equivalently, $\mathscr{E}(M) = \bigcap_{A \in \mathscr{D}(M)} \mathscr{E}'(A)$, where for A in $\mathscr{D}(M)$ we define $\mathscr{E}'(A) = \mathscr{X}(A) \cap \mathscr{R}(M)$.

It is easy to show, similarly to lemma 3.22, that $\mathscr{E}'(A) = \bigcap_{B \equiv A} \mathrm{Out}'(B)$, where

$\mathrm{Out}'(A) = \{u \in \mathscr{R}(M_0) | \forall i \leq n \ \forall v \in u \backslash A[i-1] \nexists u_i \in U_i \ u_i \prec v\}$

with $A\colon M_0 \xrightarrow{U_1} M_1 \xrightarrow{U_2} \cdots \xrightarrow{U_{n-1}} M_{n-1} \xrightarrow{U_n} M_n$

We may therefore characterize alternatively $\mathscr{E}(M)$ as the set of redex occurrences of M which are outermost in every derivation issued from M:

$$\mathscr{E}(M) = \bigcap_{A \in \mathscr{D}(M)} \mathrm{Out}'(A).$$

We shall now show that $\mathscr{E}(M)$ is not empty if M is not a normal form. But first we need a technical lemma.

NOTATION We write $\rightarrow_{\mathrm{int}}$ for $\rightarrow - \overset{\Lambda}{\rightarrow}$ (internal reduction). We say that derivation A is *internal* iff it is composed of internal reductions. That is, A in $\mathscr{D}(M)$ is internal iff $M = FM_1 \ldots M_p$ and A preserves $1, 2, \ldots, p$ (and therefore $\mathscr{F}(A) = FN_1 \ldots N_p$). Of course, if A is internal, every permutation of A is internal.

LEMMA 3.24 Let $A\colon M \rightarrow^*_{\mathrm{int}} \overset{\Lambda}{\rightarrow} N$. For any B in $\mathscr{D}(N)$ we have $\mathscr{X}(AB) = \mathscr{X}(A)$

Proof Obvious from definitions, since $\Lambda \in \mathscr{X}(B)$. ■

LEMMA 3.25 $M \in \mathscr{NF}$ iff $\mathscr{E}(M) = \varnothing$.

Proof If M is a normal form, $\mathscr{R}(M) = \varnothing$ and thus $\mathscr{E}(M) = \varnothing$. Conversely, by induction on M. Assume $M \notin \mathscr{NF}$. We show $\mathscr{E}(M) \neq \varnothing$.

Case 1: Every A in $\mathscr{D}(M)$ is internal. Thus $M = F(M_1, \ldots, M_p)$ and for some $k \leq p$, $M_k \notin \mathscr{NF}$. By induction hypothesis there exists u_k in $\mathscr{E}(M_k)$, and therefore ku_k is in $\mathscr{E}(M)$.

Case 2: There is some $A\colon M \rightarrow_{\mathrm{int}} \overset{\Lambda}{\rightarrow}$. Let us show that $\mathscr{E}(A) \subseteq \mathscr{E}(M)$. Contrariwise, assume that there exists B in $\mathscr{D}(M)$ and u in $\mathscr{E}(A) - \mathscr{X}(B)$. Since $B \sqsubseteq A \sqcup B$, we have $u \notin \mathscr{X}(A \sqsubseteq B)$ by lemma 3.11. But $\mathscr{X}(A \sqcup B) = \mathscr{X}(A)$ by lemma 3.24, a contradiction with $u \in \mathscr{E}(A)$. Therefore, $\mathscr{E}(A) \subseteq \mathscr{E}(M)$, and since $A \not\equiv 0$, we get $\mathscr{E}(M) \neq \varnothing$ by lemma 3.15. ■

This permits us to define the notion of *normal derivation issued from a term M*, similarly to what happens in λ-calculus:

DEFINITION If $M \in \mathscr{NF}$, the *normal derivation issued from M* is 0. Otherwise, let u be the leftmost occurrence in $\mathscr{E}(M)$ and A the elementary derivation: $M \overset{u}{\rightarrow} N$. The *normal derivation issued from M* is A followed by the normal derivation issued from N.

THEOREM 3.26: NORMAL DERIVATION THEOREM If M admits a normal form N, the normal derivation issued from M will end in N; it is the standard in the class of all derivations going from M to N.

Proof Let M be a term admitting a normal form N and A be some derivation from M to N. Let now B be any derivation issued from M. We have $B \sqsubseteq A$ and thus $\mathscr{X}(A) \subseteq \mathscr{X}(B)$ by lemma 3.11. This shows that $\mathscr{X}(M) = \mathscr{X}(A)$: the external occurrences of a term admitting a normal form are the occurrences external for any derivation going to this normal form. Assume now there is some u in $(\mathscr{X}(A) \cap \mathscr{R}(M)) - \mathscr{R}(A)$, i.e., u is a redex occurrence in M external to A but not contributing to A. According to lemma 3.12 we have $u \backslash A = \{u\}$, contrary to the hypothesis that N is in normal form. This shows that $\mathscr{E}(M) = \mathscr{E}(A)$. With a simple induction on $|A|$, it is now easy to show that st(A) is the normal derivation issued from M. ■

The notion of normal derivation may be considered as the generalisation to term rewriting systems of the call by name computation rule for recursive equations [2, 6, 8, 27]. In the terminology of [24], we have that all our systems are d-outer, using the dominance ordering d defined by "$u(dM)v$ iff either $u \preceq v$ or $u|v$, $u \in \mathscr{X}(M)$, and $v \notin \mathscr{X}(M)$."

In the next section, we are going to extend these results to define a call-by-need computation rule for terms possessing a normal form. Before that, let us remark that it is possible to extend lemma 3.24 to external redex occurrences, yielding a canonical form for the standard of external derivations issued from a given term.

LEMMA 3.27 Let $A: M \to_{\text{int}}^{*} \overset{\Lambda}{\to} N$. For any B in $\mathscr{O}(N)$ we have $\mathscr{E}(AB) = \mathscr{E}(A)$.

Proof Let $u \in \mathscr{R}(AB) - \mathscr{R}(A)$. We have $u \notin \text{Out}(AB)$ and therefore $u \notin \mathscr{E}(AB)$. ■

LEMMA 3.28 Let $A: M \to_{\text{int}}^{*} \overset{\Lambda}{\to} N$. For any B in $\mathscr{D}(N)$, AB is standard iff A and B are standard.

Proof An easy induction on $|A|$, using lemma 3.27 above. ■

COROLLARY Let $A: M \to_{\text{int}}^{*} \overset{\Lambda}{\to} N$ be standard. For every noninternal B in $\mathscr{D}(N)$ we have st$(B) = AC$, with $C = \text{st}(B \backslash A)$.

Proof If B is not internal, it is easy to show that $A \sqsubseteq B$. Consider $C = \text{st}(B \backslash A)$. Since A is standard, AC is standard by lemma 3.28. But $C \equiv B \backslash A$ implies then that $AC \equiv B$. By unicity of standard derivations, we get st$(B) = AC$. ■

3.3 Call by Need

We are interested in defining an interpreter for any TRS Σ in our class. Such an interpreter will be defined by a *computation rule*, where we choose in any term M some redex occurrence to contract next. The interpreter will compute correctly if, started on any term M possessing a normal form N, it defines a derivation that terminates in N after a finite number of steps.

Note that this question is non-trivial only if for some rule $\alpha \to \beta$ in Σ we have $\mathscr{V}(\alpha) - \mathscr{V}(\beta) \neq \varnothing$, since otherwise any computation rule is correct; we leave this easy proof to the reader.

In the general case, we saw in the last section that the normal computation rule (leftmost of $\mathscr{E}(M)$) is correct. Actually, with the help of lemma 3.16 we could easily have proven correct a more general computation rule: contract any redex occurrence in $\mathscr{E}(M)$. Intuitively, the redexes named by $\mathscr{E}(M)$ *need* to be contracted in order to get to the normal form of M. We shall here formalize this concept, and generalize the results of 3.2 by defining precisely what is a correct call-by-need interpreter.

DEFINITION 3.29 If M admits a normal form N, then $u \in \mathscr{R}(M)$ is a redex occurrence *needed for the normal form*, in symbols $u \in \mathscr{N}(M)$ iff $u \in \mathscr{R}(A)$ for every $A: M \xrightarrow{*} N$.

In order to prove further properties of needed redexes, consider first a few technical lemmas about outside-in derivations. In the following, we abbreviate outside-in as "oi."

LEMMA 3.30 If $A \equiv B$ with A oi, then $\mathscr{R}(A) \subseteq \mathscr{R}(B)$.

Proof Let $A = A_1 A_2$ with $A_1: M \xrightarrow{u} M'$. Then $u \in \mathscr{E}(A) = \mathscr{E}(B)$. Thus $u \in \mathscr{R}(B)$. Furthermore, $\mathscr{R}(A_2) \subseteq \mathscr{R}(B \backslash A_1)$, by the induction hypothesis. Therefore, $\mathscr{R}(A) \subseteq \mathscr{R}(B)$. ∎

COROLLARY If M has the normal form N, then $\mathscr{N}(M) = \mathscr{R}(A)$, where A is any oi derivation from M to N.

LEMMA 3.31 If A is oi and $B \sqsubseteq A$, then $A \backslash B$ is oi (except for some empty steps).

Proof By induction on $|A|$. Let $A = A_1 A_2$, with $A_1: M \xrightarrow{u} N$. By the induction hypothesis, we already know that $A_2 \backslash (B \backslash A_1)$ is oi. Consider now $A_1 \backslash B$, i.e., $u \backslash B$. If $u \backslash B = \varnothing$, then $A \backslash B$ is obviously oi. Suppose now $u \backslash B \neq \varnothing$. Since A is oi and $A \equiv B \sqcup A$, we know that $u \in \mathscr{E}(A) = \mathscr{E}(A \backslash B)$. Thus $u \in \mathrm{Out}(B \sqcup A)$, by lemma 3.22, and $u \backslash B = \{u\}$. Now for all $C \equiv A \backslash B$, one has $u \in \mathrm{Out}(C)$, since $BC \equiv A$ and $u \in \mathscr{E}(A)$. Again by lemma 3.22, one has $u \in \mathscr{E}(A \backslash B)$. ∎

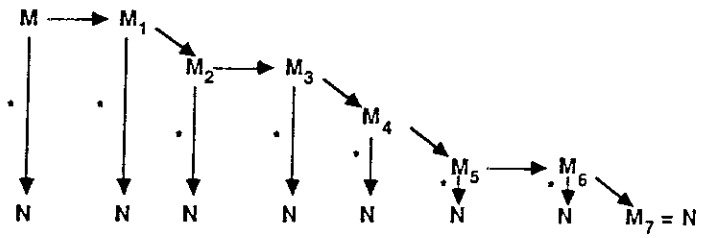
LEMMA 3.32 If A, B are oi and $A \equiv B$, then $|A| = |B|$.

Proof If $|A| = 0$, then $B \equiv 0$ and $B = 0$, since B is oi. Similarly if $|B| = 0$. Now let $A = A_1 A_2$ and $B = B_1 B_2$ with $A_1: M \xrightarrow{u} N$ and $B_1: M \xrightarrow{v} P$. Denote temporarily by C' the derivation C, where the empty steps are suppressed. As $\mathscr{E}(A) = \mathscr{E}(B)$ and $u \in \mathscr{E}(A)$, $v \in \mathscr{E}(A)$, we have u and v disjoint. Thus if $B_3 = B_1 \backslash A_1$ and $A_3 = A_1 \backslash B_1$, one has $B_3: N \xrightarrow{v} Q$ and $A_3: P \xrightarrow{u} Q$. Therefore, using the previous lemma and an induction on $|A| + |B|$, we get $|A| = 1 + |A_2| = 1 + |(B_2 \backslash A_3)'|$. Again by induction, $|(B_2 \backslash A_3)'| = |(A_2 \backslash B_3)'|$. But one also has $|B| = 2 + |(A_2 \backslash B_3)'|$. ∎

NOTATION Let $d(A) = |\mathrm{st}(A)|$.

By the previous lemma, since $\mathrm{st}(A)$ is oi, one has $d(A) = |B|$ for any B oi such that $B \equiv A$. The fact that external redexes are preserved until they are reduced gives us properties of this notion of distance.

LEMMA 3.33 If $A \sqsupseteq B$, then $d(A) \geq d(A \backslash B)$.



$$d(M) = d(M_1) > d(M_2) = d(M_3) > d(M_4) > d(M_5) = d(M_6) > d(M_7) = 0$$

Figure 8

Proof Obvious from the two previous lemmas. ■

LEMMA 3.34 If $A \sqsupseteq B$ and $|B| = 1$, then $d(A) > d(A\backslash B)$ iff $\mathscr{R}(\mathrm{st}(A)) \cap \mathscr{R}(B) \neq \varnothing$.

Proof Let $C = \mathrm{st}(A)$ and $C = C_1 C_2$, with $C_1 : M \xrightarrow{u} N$. Let $B = M \xrightarrow{v} P$. Then with $C \equiv A \sqsubseteq B$, we get $C \equiv B \sqcup C$ and $u \in \mathrm{Out}(B \sqcup C)$, since $u \in \mathscr{E}(C) = \mathscr{E}(B \sqcup C)$. This means that $u\backslash B = \varnothing$ iff $u \in V$. Now the lemma follows by induction on $|C|$, since $A\backslash B \equiv C\backslash B$ and by lemma 3.31 and lemma 3.32. ■

For the following corollary we denote by $d(M)$ the length of the standard derivation of M to its normal form when it exists.

COROLLARY Let M have the normal form N. Then $d(M) = 0$ iff $M = N$. In addition, let $M \xrightarrow{U} M'$. Then if $U \cap \mathscr{N}(M) = \varnothing$, one has $d(M) = d(M')$. Otherwise, $d(M) > d(M')$.

This is summarized in figure 8, where the slanted steps are needed and the vertical derivations are standard.

This concludes the proof of the correctness of the call-by-need computation rule. Actually, any interpreter which is fair for needed redexes, in the sense that it will never postpone forever the contraction of needed redexes, is correct for computing normal forms. Note that the standard derivation is the longest derivation contracting only needed redexes.

The next problem we shall tackle is how to effectively compute a needed redex in a given term. We know from the corollary to lemma 3.30 that $\mathscr{N}(M) = \mathscr{R}(A)$ for A any outside-in derivation going from M to its normal form. But this is useless practically. Intuitively we want our interpreter to compute an element of $\mathscr{N}(M)$ without looking ahead. This problem will be stated precisely and solved in part II of this paper (chapter 12).

References

[1] E. Ashcroft, W. Wadge. LUCID—A formal system for writing and proving programs. *SIAM J. on Computing* 5, no. 3 (1976): 336–354

[2] G. Berry, J. J. Lévy. Minimal and optimal computations of recursive programs. *JACM* 26, no. 1, 1979.

[3] R. Burstall. Design considerations for a functional programming language. Infotech State of the Art Conf., Copenhagen, 1977.

[4] R. Burstall, J. A. Goguen. Putting theories together to make specifications. 5th IJCAI Conf., Boston, 1977.

[5] R. M. Burstall, D. B. Macqueen, D. I. Sannella. HOPE: An experimental applicative language. Report CSR-62-80, Computer Science Dept., University of Edinburgh, Feb. 1981.

[6] J. M. Cadiou. Recursive definitions of partial functions and their computations. Ph.D. thesis, Stanford, March 1972.

[7] H. B. Curry, R. Feys. *Combinatory Logic*, vol. 1. North-Holland, 1958.

[8] P. J. Downey, R. Sethi. Correct computation rules for recursive languages. *SIAM Journal* 5, no. 3, Sept. 1976.

[9] M. Gordon, R. Milner, L. Morris, M. Newey, C. Wadsworth. A metalanguage for interactive proof in LCF. Report CSR-16-77, Computer Science Dept., University of Edinburgh, Sept. 77.

[10] M. J. Gordon, A. J. Milner, C. P. Wadsworth. Edinburgh LCF. Springer-Verlag Lecture Notes in Computer Science, no. 78, 1979.

[11] J. V. Guttag, E. Horowitz, D. R. Musser. Abstract data types and software validation. Research report 76-48, ISI, Aug. 1976.

[12] R. Hindley. An abstract Church-Rosser theorem. Pt. 1, *J. Symbolic Logic* 34 (1969): 545–560; Pt. 2, *J. Symbolic Logic* 39 (1974): 1–21.

[13] G. Huet. Confluent reductions: Abstract properties and applications to term rewriting systems. *JACM* 27, no. 4 (1980): 797–821.

[14] G. Huet, J. J. Lévy. Computations in regular rewriting systems, II. Chapter 12, this volume.

[15] G. Kahn, G. Plotkin. Domaines concrets. Rapport Laboria no. 336, IRIA, December 1978.

[16] S. C. Kleene. *Introduction to Metamathematics*. North-Holland, 1952.

[17] D. E. Knuth, P. B. Bendix. Simple word problems in universal algebras. In *Computational Problems in Abstract Algebra*, ed. J. Leech. Pergamon Press, 1970, pp. 263–297.

[18] J. J. Lévy. Reductions correctes et optimales dans le lambda-calcul. Thesis, Université Paris 7, Jan. 1978.

[19] J. McCarthy. Recursive functions of symbolic expressions and their computation by machine, part I. *CACM* 3 (April 1960): 184–195.

[20] J. McCarthy. A basis for a mathematical theory of computation. In *Computer Programming and Formal Systems*, ed. P. Braffort and D. Hirschberg. North-Holland, 1963, pp. 33–70.

[21] R. Milner. Implementation and application of Scott's logic for computable functions. *Proc. ACM Conf. on Proving Assertions about Programs*. SIGPLAN notices 7 1, Las Cruces, Jan. 1972.

[22] D. R. Musser. A data type verification system based on rewrite rules. 6th Texas Conf. on Computing Systems, Austin, Nov. 1978.

[23] M. Nivat. On the interpretation of recursive polyadic program schemes. *Symposia Mathematica,* vol. 15. Instituto Nazionale di Alta Matematica, Italy, 1975, pp. 225–281.

[24] M. O'Donnell. *Computing in Systems Described by Equations*. LNCS no. 58, Springer-Verlag, 1977.

[25] J. C. Raoult, J. Vuillemin. Operational and semantic equivalence between recursive programs. 10th SIGACT Conf., San Diego, 1978.

[26] B. K. Rosen. Tree manipulation systems and Church-Rosser theorems. *JACM* 20, no. 1, 1973.

[27] J. Vuillemin. Correct and optimal implementation of recursion in a simple programming language. *JCSS* 9, no. 3 (1974): 332–354.