JOURNAL OF COMPUTER AND SYSTEM SCIENCES **32**, 323–362 (1986)

Conditional Rewrite Rules: Confluence and Termination

J. A. BERGSTRA

University of Amsterdam and State University of Utrecht, The Netherlands

AND

J. W. KLOP

Centre for Mathematics and Computer Science, Amsterdam, The Netherlands

Received September 6, 1982; revised June 11, 1985

Algebraic specifications of abstract data types can often be viewed as systems of rewrite rules. Here we consider rewrite rules with conditions, such as they arise, e.g., from algebraic specifications with positive conditional equations. The conditional term rewriting systems thus obtained which we will study, are based upon the well-known class of left-linear, non-ambiguous TRSs. A large part of the theory for such TRSs can be generalized to the conditional case. Our approach is non-hierarchical: the conditions are to be evaluated in the same rewriting system. We prove confluence results and termination results for some well-known reduction strategies.

INTRODUCTION

This paper is concerned with term rewriting systems involving conditional rewrite rules. Such systems arise in a natural way from algebraic data type specifications using positive conditional equations, but may just as well appear in a different context. Our aim is to provide a self-contained introduction in the subject covering various topics, such as: confluence, reduction strategies and termination, and decision algorithms for normal forms.

While working in this subject we received Pletat, Engels, and Ehrich [18]. This paper has had a considerable influence on our ideas, leading us, however, to a different proposal for the semantics of conditional rewrite rules, avoiding hierarchies, but introducing circularities that turn out to be not problematic in the end.

For some related work on conditional TRSs we refer to [6, 12, 13].

We will now give a survey of the paper. We will consider systems Σ of *positive conditional equations*, as they are called in [7], which have the form

$$t_1 = s_1 \wedge \cdots \wedge t_n = s_n \Rightarrow t = s,$$

for some $n \geqslant 0$. Here the $t_i = s_i$ $(i = 1,\dots, n)$ and $t = s$ are equations, possibly containing varables. Such systems arise for instance in algebraic semantics as specifications of abstract data types, see [7]. If Σ is a system of positive conditional equations, Σ_u will be the "unconditional part" of Σ, that is the set of equation schemes obtained by removing the conditions (i.e., the LHSs of the implications). The system of equation schemes Σ_u can be made into a term rewriting system (TRS), by choosing a direction of rewriting: $t \to s$. Often this direction is clearly suggested by the equation $t = s$. Now we will impose (just as in [18]) the restriction that the TRS Σ_u is *non-ambiguous and left-linear*. For such TRSs, which we will call of type 0 in this paper, the syntactical theory is well developed; cf. [3, 9, 10, 11, 14, 16, 17].

While it is clear how to associate a TRS to a system of equation schemes (anyway in the case we are considering), this is less clear in the presence of these conditions:

I. One possibility is to consider "conditional reduction rule schemes" of the form

$$t_1 = s_1 \wedge \cdots \wedge t_n = s_n \Rightarrow t \to s.$$

Such conditional reduction rule schemes will be called of type I. Likewise a TRS is of type I if it contains only reduction rules of type I.

II. Another possibility is to consider conditional rules of the form

$$t_1 \downarrow s_1 \wedge \cdots \wedge t_n \downarrow s_n \Rightarrow t \to s$$

where "$\downarrow$" denotes "having a common reduct."

III. Third, one could consider

$$t_1 \twoheadrightarrow s_1 \wedge \cdots \wedge t_n \twoheadrightarrow s_n \Rightarrow t \to s,$$

where $\twoheadrightarrow$ is the transitive reflexive closure of the one step reduction relation generated (in a sense made precise below) by these schemes.

It turns out that this last possibility yields in general not a confluent reduction (i.e., having the Church–Rosser property). A "better" type of conditional reduction rule is:

III_n. $t_1 \twoheadrightarrow n_1 \wedge \cdots \wedge t_k \twoheadrightarrow n_k \Rightarrow t \to s$, where the n_i, $i = 1,..., k$, are closed normal forms in the sense of the unconditional Σ_u.

Now in all cases I, II, $\mathrm{III}_{(n)}$ there is an obvious circularity involved in the definition of the reduction relation $\to$. In [18] this problem is solved by means of an hierarchical approach: the conditions (which are there of type III_n, to be precise: of the form $t_i \twoheadrightarrow true$) must be evaluated on a lower level of the hierarchy. Here we will not suppose such a hierarchical structure of the TRSs, and define the reduction relation ($\to$) by a "least fixed point" construction; for type I and III_n reductions we can then prove confluence. That is, the circularity is harmless in case III_n, and also for type I. In fact, the whole syntactical theory for type 0 carries over without effort to type I and III_n, including termination criteria. However, a major problem with the conditional TRSs is that the set of normal forms and the set of redexes need not be decidable.

For type III in general it is not surprising to see that such reductions need not be confluent, for, it is not clear that a condition $t_i \twoheadrightarrow s_i$ is "stable" under reductions. For type II it does seem reasonable to conjecture confluence; but we will show that in fact this conjecture is false. The case of type I is very easy.

The really interesting case is III_n. We will show that III_n-reductions are confluent, and have in general all desirable properties of 0-reductions, including termination (when possible) of reduction strategies like full substitution (or full computation), leftmost reduction, parallel outermost reductions. Most of these results are already obtained in [18], however, for the "hierarchical" III_n-TRSs.

Note that we have not placed restrictions on the conditions $t_i = s_i$ (type I) or $t_i \twoheadrightarrow n_i$ (type III_n), other than the unconditional normal form requirement (which can be immediately checked by looking only at the LHSs t of the RHSs $t \to s$ of the conditional rules) in III_n. This is intended: the $t_i = s_i$ or $t_i \twoheadrightarrow n_i$ may have other variables than the ones in $t = s$. E.g., the rule (as in the definition of an equivalence relation)

$$E(x, y) \twoheadrightarrow true \text{ and } E(y, z) \twoheadrightarrow true \Rightarrow E(x, z) \to true$$

is allowed.

On the other hand, an unconditional rule like

$$E(x, x) \to true$$

will not be allowed here, since we stipulated that the unconditional part Σ_u of the TRSs Σ we will consider, must be of type 0. Let us call a TRS Σ' of type 0′ if it can be obtained from a type 0 TRS Σ by identifying some variables in the LHSs of the rule schemes.

Now we give a translation of type III_n systems into type 0 and of type II into type 0′. We do not, however, explore the formal aspects of this translation and use it

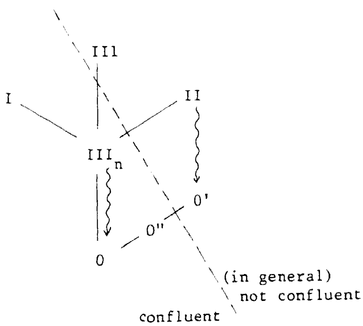

FIGURE 1

mostly as a heuristic tool to show that type II and III reductions are in general not confluent.

A survey of the confluence results is given in Fig. 1, where an upward line means that a TRS of the lower type is also a TRS of the higher type. The central point in this diagram, type III_n, will also be a focus of our interest in this paper. The wavy downward arrows refer to the "translation" mentioned above and given in Section 2.5. Type 0″ is a subtype of 0′, obtained by stipulating that the "non-linear" operators may not occur in the RHSs of the rule schemes (Sect. 1.5).

We have included an Appendix devoted to O'Donnell's theorem that "eventually outermost" reductions (including the parallel outermost reductions) must terminate when possible, and likewise for leftmost reductions in the case of left-normal rules. In fact, we prove a stronger version, applying also to the case of term rewriting systems with *bound variables*, such as λ-calculus. Indicating the presence of bound variables with "*," all our results except Theorem 5.4 generalize from 0 to 0*, I to I*, III_n to III_n^*. Since bound variables are not the main topic of this paper, we have separated this proof in an Appendix so that it can easily be omitted (or, singled out). Type 0* reductions systems are called "regular combinatory reduction systems" in [14], where "regular" means "non-ambiguous and left-linear."

The structure of the sequel of this paper is as follows:

1. Preliminaries. 2. Conditional Term Rewriting Systems. 3. Confluence. 4. Complexity of Normal Forms. 5. Termination. 6. Possible Extensions. 7. Appendix: Parallel Outermost and Leftmost Reductions.

1.1 *Term Rewriting Systems*

We will briefly introduce the well-known notion of a term rewriting system (TRS), as studied, e.g., in [3, 9, 10, 11, 14, 16, 17]. First we will consider unconditional TRSs.

A term rewriting system Σ is a triple $\langle \mathscr{F}, \mathscr{V}, \mathbb{R} \rangle$ where $\mathscr{F}$ is a set of *ranked operators*, i.e., each $F \in \mathscr{F}$ has an *arity* which is the number of arguments F is supposed to act upon. The arity may be 0, in which case F is also called a constant. $\mathscr{V}$ is a set of *variables*, necessary to describe the set of *reduction rule schemes*, $\mathbb{R}$. A reduction rule scheme, or rule scheme for short, is a pair (t, s), written as $t \to s$, where $t, s \in \mathrm{Ter}(\Sigma)$, the set of terms built from $\mathscr{F}$ and $\mathscr{V}$. So $\mathbb{R}$ is a binary relation on $\mathrm{Ter}(\Sigma)$. The set of closed Σ-terms, $\mathrm{Ter}^c(\Sigma)$, contains only terms without variables $a, b, c, \ldots, x, y, z \in \mathscr{V}$. We will use t, s for terms, but sometimes also $M, N, \ldots$. An *instantiation* ρ is a map $\mathscr{V} \to \mathrm{Ter}^c(\Sigma)$. If $t \in \mathrm{Ter}(\Sigma)$, then $\rho(t)$ denotes the result of simultaneous substitution of $\rho(x)$ for every occurrence of x in t.

$\bar{\mathbb{R}}$ is the set of all *closed instances* obtained from the rule schemes $\mathbb{R}$; i.e., if $t \to s \in \mathbb{R}$ then $\rho(t) \to \rho(s) \in \bar{\mathbb{R}}$ for all ρ. The elements of $\bar{\mathbb{R}}$ are called *closed rules*; we will drop the word "closed" sometimes. The LHSs of the rules are called *redexes*; $\mathrm{RED}(\Sigma)$ is the set of all redexes of Σ. A term without redexes as subterms is a *normal form*; $\mathrm{NF}(\Sigma)$ is the set of normal forms.

A *context* $C[\ ]$ is a term with one "hole." More precisely: let $\square$ be a "fresh" variable. Then a context $C[\ ]$ is a term containing exactly one occurrence of $\square$. (The trivial context is $\square$ itself.) We write $C[t]$ to denote the result of substituting t in the open place $\square$.

If R is a binary relation on $\mathrm{Ter}^c(\Sigma)$, then R^m will be the "*contextual closure*" of R, defined by

$$(t, s) \in R \Rightarrow (C[t], C[s]) \in R^m \qquad \text{for all} \quad C[\ ].$$

R^* is, as usual, the transitive reflexive closure of R. For notational ease, we write $R^\circ = (R^m)^*$. Note that $\varnothing^\circ = \equiv$, syntactical equality.

If the infix notation $t \to s$ is used, the relation $\to$ will be called "reduction" and instead of $\to^\circ$ we use the notation $\twoheadrightarrow$ (which is easier to use in reduction diagrams).

Remark on notation. Terms are notated by $t, s, ...$ as well as by $M, N, ...$ We apologize for this inconvenience.

1.2. *Applicative vs. Ranked TRSs; TRSs with Many-Sorted Signature*

As we have introduced TRSs in 1.1, each operator has a fixed arity and term formation is otherwise unrestricted. In practice however, we will often deal with TRSs having a (many-sorted) signature, as in Example 2.3(i). This concept is standard in the literature, and we will not give a definition here. See, e.g., [11]. Nowhere, however, in this paper will the concept of signature play a role; that is, everything works out for TRSs with signature exactly as for TRSs without signature restrictions, i.e., one-sorted.

Instead of *ranked* TRSs (i.e. each operator has a fixed arity), one can also consider *applicative* TRSs. The prime example of such a TRS is combinatory logic (CL) as in [1, 5], with basic operators S, K, I and terms $M \in \mathrm{Ter}(\mathrm{CL})$ given by the inductive definition

$M ::= I, K, S/(M_1 M_2)$, and reduction rules schemes

$$Sxyz \to xz(yz)$$

$$Kxy \to x$$

$$Ix \to x$$

(here the convention of bracket association to the left is used). An applicative system Σ can easily be viewed as a ranked system Σ_A, by introducing a binary operator $A(\ ,\ )$ and considering S, K, I as 0-ary operators (constants). Then the rules of CL_A are:

$$A(A(A(S, x), y), z) \to A(A(x, z), A(y, z))$$

$$A(A(K, x), y) \to x$$

$$A(I, x) \to x.$$

Vice versa, a ranked TRS Σ_r can be viewed as a "sub-TRS" of an applicative TRS Σ; e.g., if $\Sigma_r = \{C, P(x, Q(y)) \to Q(x)\}$ then Σ_r is a "sub-TRS" of Σ (see 1.4.0), where Σ has terms defined by $M ::= C, P, Q/(M_1 M_2)$ and the rule $Px(Qy) \to Qx$. So the terms of (an isomorphic copy of) Σ_r would be given by

$$M ::= C/PM_1M_2/QM.$$

In fact, we may use TRSs which are partly applicative and partly ranked; e.g.,

$$\mathrm{CL} + D(x, x) \to I.$$

At one point, however, there is a crucial difference between ranked and applicative TRSs, namely in the formulation of a theorem about non-linear TRSs, see 1.5.2.2.

1.3. *Regular Reductions*

An important class of reduction systems is the class of *regular* TRSs $\Sigma = \langle \mathscr{F}, \mathscr{V}, \mathbb{R} \rangle$. Here the rule schemes in $\mathbb{R}$ are subject to the following conditions:

(i) if $t \to s \in \mathbb{R}$, the leading symbol of t is an operator $\in \mathscr{F}$ (so $t \notin \mathscr{V}$);

(ii) if $t \to s \in \mathbb{R}$, then the variables in s occur already in t;

(iii) if $t \to s \in \mathbb{R}$, then t is *linear*, i.e., no variable occurs more than once in t. (The rule scheme $t \to s$ is called left-linear if t is linear.)

(iv) if $\mathbb{R} = \{t_i \to s_i \mid i \in I\}$ then the rule schemes do not "interfere," i.e., they are *non-ambiguous*. One also says that $\mathbb{R}$ has the *non-overlapping* property. This property is defined as follows.

1.3.1. Definition. Let $\mathbb{R} = \{r_i \mid i \in I\}$, where $r_i = t_i \to s_i$ be the set of rule schemes of a TRS Σ. We may suppose that $\mathbb{R}$ contains no rule schemes which can be obtained from each other by renaming of variables. Σ is called a *non-ambiguous* (or *non-overlapping*) TRS iff the following holds:

(i) if the r_i-redex $\rho(t_i)$ contains the r_j-redex $\rho'(t_j)$, where $i \neq j$ and ρ, ρ' are some instantiations, then the redex $\rho'(t_j)$ is already contained by $\rho(x)$ for some variable x occurring in t_i;

(ii) if the r_i-redex $\rho(t_i)$ contains the r_i-redex $\rho'(t_i)$ for some ρ, ρ', then either $\rho(t_i) \equiv \rho'(t_i)$ or $\rho'(t_i)$ is already contained by $\rho(x)$ for some variable x occurring in t_i.

Here "r is contained in t" means that r is a subterm of t, notation $r \subseteq t$. Equivalently: $t \equiv C[r]$ for some context $C[\ ]$ of r. So, in a well-known terminology, non-ambiguity means that there are no critical pairs.

1.3.2. Examples. (i) $\mathbb{R} = \{P(Q(x)) \to R(x),\ Q(R(x)) \to S\}$ is ambiguous by clause (i) of Definition 1.3.1;

(ii) $\mathbb{R} = \{P(P(x)) \to P(x)\}$ is ambiguous by clause (ii);

(iii) $\mathbb{R} = \{D(x, x) \to E, \ldots\}$ yields a nonregular TRS since the displayed rule scheme is not left-linear.

1.3.3. *Remark.* It is possible to be slightly more liberal in the definition of ambiguity, without losing any of the properties of regular reductions. (This definition is adopted by O'Donnell [17].) Namely, define $\Sigma = \langle \mathscr{F}, \mathscr{V}, \mathbb{R} \rangle$ (where $\mathbb{R} = \{r_i \mid i \in I\}$, $r_i = t_i \to s_i$) is a *weakly non-ambiguous* TRS iff the following holds:

(i) if the r_i-redex $\rho(t_i)$ contains the r_j-redex $\rho'(t_j)$, where $i \neq j$ and ρ, ρ' are some instantiations, then the redex $\rho'(t_j)$ is either

(a) already contained by $\rho(x)$ for some x in t_i or

(b) $\rho(t_i) \equiv \rho'(t_j)$ and $\rho(s_i) \equiv \rho'(s_j)$. (I.e. the rules $\rho(r_i)$ and $\rho'(r_j)$ coincide.)

(ii) as in Definition 1.3.1.

Note that non-ambiguity of Σ depends only of the LHSs t_i of the rule schemes in $\mathbb{R}$, while for weak non-ambiguity also the RHSs s_i must be considered.

An example of a set of weakly non-ambiguous rule schemes, which is ambiguous, is given by the "parallel or" rule schemes:

$$or\ (true, x) \to x$$

$$or\ (x, true) \to x.$$

Let us call a TRS which is leftlinear and weakly non-ambiguous, a *weakly regular* TRS. Then the theory for regular TRSs as, e.g., in [14], on which most of the sequel is based, seems to carry over without problems to weakly regular TRSs. We will stick to regular TRSs as the basis for the sequel, however.

1.4. *Reduction Diagrams for Regular Reductions*

Let Σ be a regular TRS. Then, as is well known, $\Sigma \models \text{CR}$. (Σ has the Church–Rosser property.) I.e.: if $\mathscr{R}_1 = t_0 \to t_1 \to \cdots \to t_n$ and $\mathscr{R}_2 = t_0 \to t'_1 \to \cdots \to t'_m$ are two "divergent" reductions of $t_0 \in \text{Ter}(\Sigma)$, then there are "convergent" reductions $\mathscr{R}_3 = t_n \to \cdots \to s$ and $\mathscr{R}_4 = t'_m \to \cdots \to s$. Instead of saying that Σ has the CR-property, we will also say that Σ-reductions are confluent.

A stronger version of the CR-theorem for regular TRSs asserts that convergent reductions $\mathscr{R}_3, \mathscr{R}_4$ can be found in a canonical way, by adjoining "*elementary diagrams*" as suggested in Fig. 2. In this way the *reduction diagram* $\mathscr{D}(\mathscr{R}_1, \mathscr{R}_2)$ originates, and in [14] it is proved that the construction terminates and yields $\mathscr{R}_3, \mathscr{R}_4$ as desired. It is fairly evident how to define the elementary diagrams; e.g., if $\Sigma = \text{CL}$ as in 1.2, then the following are examples (see Fig. 3). Here "$\varnothing$" denotes an "empty" or "trivial" step, necessary to keep the reduction diagram in a rectangular

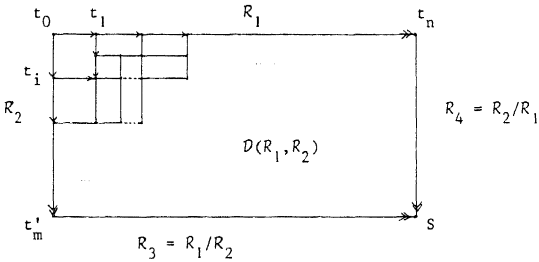

FIGURE 2

shape. $\emptyset$-steps also occur in elementary diagrams of the form, e.g. (see Fig. 4). The reduction $\mathcal{R}_3$ constructed above in $\mathcal{D}(\mathcal{R}_1, \mathcal{R}_2)$ is called the *projection* of $\mathcal{R}_1$ by $\mathcal{R}_2$, written: $\mathcal{R}_3 = \mathcal{R}_1/\mathcal{R}_2$. Similarly $\mathcal{R}_4 = \mathcal{R}_2/\mathcal{R}_1$.

1.4.0. *Sub-TRSs*

Up to here we have only considered regular TRSs $\Sigma = \langle \mathcal{F}, \mathcal{V}, \mathbb{R} \rangle$, where term formation is unrestricted. However, since most of the relevant properties of regular TRSs derive from the notion of reduction diagram, it is sensible to enlarge the class of regular TRSs such that they include also "sub-TRSs" Σ' of Σ, defined as follows:

Let $T \subseteq \text{Ter}(\Sigma)$ be such that T is closed w.r.t. elementary diagrams. (I.e., if $t_0, t_1, t_2 \in T$ such that $t_0 \to t_1$, $t_0 \to t_2$ then all terms involved in $\mathcal{D}(t_0 \to t_1, t_0 \to t_2)$ are in T.) Then the restriction Σ' of Σ to T is called a *sub*-TRS of Σ. We write $\Sigma' \sqsubseteq \Sigma$.

So, in the sequel a regular TRS may be either a "full" TRS where term formulation is unrestricted or a sub-TRS of a "full" TRS. This means that TRSs, where term formation is restricted by signature requirements are also in our scope.

The next three subsections 1.4.1,..., 1.4.3 are preliminaries only for the Appendix.

1.4.1. *The Parellel Moves Lemma*

Let $\mathcal{R}$ be a Σ-reduction $t_0 \to \cdots \to t_n$ and let $s \subseteq t_0$ be a redex. Contraction of redex s (i.e., replacing s by its one step reduct) will be displayed (sometimes) by the notation $t_0 \to^s t'_0$. Now consider $\mathcal{D}(t_0 \to^s t'_0, \mathcal{R})$ (in Fig. 5). Then the reduction $\mathcal{R}'$ (the projection of reduction step $t_0 \to^s t'_0$ by $\mathcal{R}$) consists of a reduction of all the "*descendants*" of s via $\mathcal{R}$.

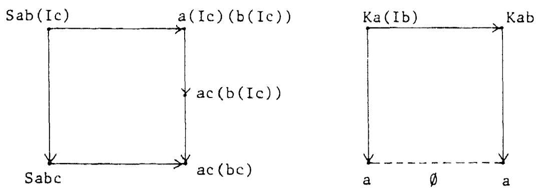

FIGURE 3

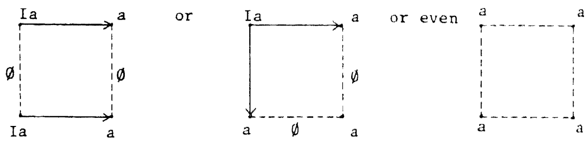

FIGURE 4

1.4.1.1. *Descendants.* The notion of "*descendant (via $\mathscr{R}$)*" is defined as follows:

(i) If $t \to s$ is a rule scheme and $\rho(t) \to \rho(s)$ an instantiation such that $t' \subseteq \rho(x)$ for some occurrence of a variable x in t, then t' gives rise to some copies, called *descendants* of t', in $\rho(s)$, depending on the possible occurrences of x in s.

(ii) Furthermore, if $C_1[C_2[\rho(t)]] \to C_1[C_2[\rho(s)]]$, where $C_2[\ ]$ is not the trivial context (i.e., $\rho(t) \subsetneq C_2[\rho(t))]$), then $C_2[\rho(s)]$ is the (unique) descendant of $C_2[\rho(t)]$.

Notation. If $M \to N$ is a reduction step, $A \subseteq M$, $B \subseteq N$ then $A - \cdot - \cdot \to B$ means "B is a descendant of A."

1.4.1.2. *Remark.* If B is a descendant of A, A is also called an *ancestor* of B. Descendants of redexes are also called *residuals.* Note that the *contractum* $\rho(s)$ of a redex $\rho(t)$ is not a descendant of $\rho(t)$.

If in (ii) $C_2[\ ]$ is allowed to be the trivial context, the resulting notion will be that of "*quasi-descendants.*" So the contractum of a redex is a quasi-descendant of that redex.

Note that residuals of a r_i-redex are again r_i-redexes. Furthermore, note that in the above reduction diagram, $\mathscr{R}'$ consists of a construction of *disjoint* residuals $s_1, s_2,\ldots$ of s. (This would not be the case in the presence of bound variables as in λ-calculus.)

1.4.2. *Equivalent Reductions*

The very useful notion of "*equivalence of reductions*" was introduced first in [15]. Intuitively, two reductions $\mathscr{R}_1, \mathscr{R}_2$, both from t to t', are equivalent (written $\mathscr{R}_1 \cong \mathscr{R}_2$) when the "same" reduction steps are performed but possibly in a permuted order. Since redexes may be nested and contraction of one redex may mul-

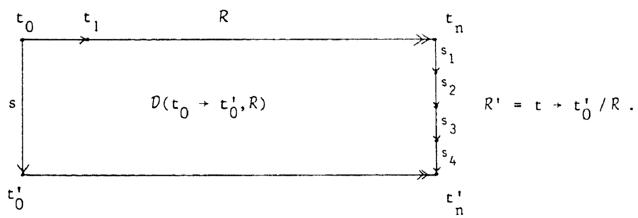

FIGURE 5

tiply subredexes, it is not quite clear what "permuted" means; but via the notion of reduction diagram this can be made precise:

$$\mathscr{R}_1 \cong \mathscr{R}_2 \Leftrightarrow \mathscr{R}_1/\mathscr{R}_2 = \mathscr{R}_2/\mathscr{R}_1 = \varnothing$$

(so $\mathscr{D}(\mathscr{R}_1, \mathscr{R}_2)$ has empty right and lower sides).

1.4.3. *Finite Developments*

Let t be a Σ-term and let $\mathbb{R}$ be a set of redex occurrences in t. Then a reduction of t in which only residuals of redexes in $\mathbb{R}$ are contracted, is called a *development* (of t w.r.t. $\mathbb{R}$). It is not hard to prove that every development of t w.r.t. $\mathbb{R}$ must be finite (see, e.g., [3, 14, 16]).

A development $t_0 \to \cdots \to t_n$ of t_0 w.r.t. $\mathbb{R}$ is called *complete* if it cannot be prolonged (i.e., in t_n there are no residuals of redexes in $\mathbb{R}$ left). All complete developments of t w.r.t. $\mathbb{R}$ and in the same result. We even have

1.4.3.1. PROPOSITION. *All complete developments of t* w.r.t. $\mathbb{R}$, *a set of redex occurrences in t, are equivalent.*

For a proof, see e.g., [14]. ▮

1.5. *Nonlinear Reductions*

1.5.1. *Type* 0′*TRSs*

For the purpose of a classification to be used in this paper, we will call a regular TRS to be of *type* 0. We will in the sequel briefly be concerned with a class of TRSs which will be called to be of type 0′ and which is obtained as follows from type 0 TRSs.

Let $\Sigma = \langle \mathscr{F}, \mathscr{V}, \mathbb{R} \rangle$ be a TRS of type 0. Let Σ' be a TRS$\langle \mathscr{F}, \mathscr{V}, \mathbb{R}' \rangle$ whose set of rule schemes $\mathbb{R}'$ is obtained from $\mathbb{R}$ by identifying some of the variables occurring in the rule schemes which were previously different. So Σ' is no longer left-linear.

1.5.1.0. EXAMPLE. Σ has set of rule schemes $\mathbb{R} = \{D(x, y) \to E,\ C(x) \to D(x, C(x)),\ B \to C(B)\}$. Identifying x, y we obtain Σ' with rule schemes $\mathbb{R} = \{D(x, x) \to E,\ C(x) \to D(x, C(x)),\ B \to C(B)\}$.

Now Σ is of type 0, and hence $\Sigma \models \mathrm{CR}$. However, for the 0′ TRS Σ' the CR property does not hold; for, consider $CB \to D(B, CB) \to D(CB, CB) \to E$ and $CB \to C(CB) \to C(D(B, CB)) \to C(D(CB, CB)) \to C(E)$. Then $C(E)$, E have no common reduct, as can easily be proved.

1.5.2. *Type* 0″ *TRSs*

Now let $\Sigma = \langle \mathscr{F}, \mathscr{V}, \mathbb{R} \rangle$ be a TRS of type 0′.

1.5.2.0. DEFINITION. (i) Let $t \to s \in \mathbb{R}$ be a non-leftlinear rule scheme. Let P be the leading symbol of t. Then P is called a *nonlinear operator.*

(ii) Now suppose that Σ is a *ranked* TRS of type $0'$. Then Σ is called of type $0''$ if none of its nonlinear operators occurs in a RHS of some rule scheme in $\mathbb{R}$.

The following theorem is a corollary of a result in [14], as noted by [5].

1.5.2.1. THEOREM. *Let Σ be of type $0''$. Then $\Sigma \models$ CR.*

1.5.2.2. *Remark.* The hypothesis that Σ is ranked in Definition 1.5.2.0 (ii) is essential for the confluency of $0''$-reductions. For, consider $\Sigma = \text{CL}$ (as in 1.2) augmented by the rule $Dxx \to E$. Then, as demonstrated in [14], the counterexample to CR for Σ' in Example 1.5.1.0 can be simulated for the present $\Sigma = \text{CL} + Dxx \to E$. Yet the only nonlinear operator D in Σ occurs in no RHS of a rule scheme.

Translating Σ to a ranked TRS Σ_A, we get the rule schemes of CL_A (see 1.2) augmented by $A((A(D, x), x) \to E$. Now A is the nonlinear operator (not D) and indeed A occurs in several RHSs of rule schemes of Σ_A, as has to be the case since $\Sigma \not\models$ CR implies evidently that also $\Sigma_A \not\models$ CR.

2. CONDITIONAL TERM REWRITING SYSTEMS

Algebraic specifications of abstract data types often contain not only equation schemes $t(\mathbf{x}) = s(\mathbf{x})$ (which can be modeled by reduction schemes $t(\mathbf{x}) \to s(\mathbf{x})$), but also conditional equation schemes $\mathbb{Q}(\mathbf{x}) \Rightarrow t(\mathbf{x}) = s(\mathbf{x})$, where $\mathbb{Q}$ is some predicate of the variables $\mathbf{x}$. Indeed, conditional reduction rule schemes of the form $\mathbb{Q}(\mathbf{x}) \Rightarrow t(\mathbf{x}) \to s(\mathbf{x})$ are considered in [16]. There some "well-behavior" of the $\mathbb{Q}(\mathbf{x})$ is explicitly required in order to have confluence and other properties of the generated reductions.

We will consider reduction rule schemes such as they can be associated to what is called in [7] *positive conditional equations.* These are of the form

$$t_1 = s_1 \wedge \cdots \wedge t_n = s_n \Rightarrow t = s \qquad (*)$$

where t_i, s_i $(i = 1, \ldots, n)$ and t, s are open terms. The basic assumption that we will make (just as in [16, 18]) to deal with positive conditional equation schemes, is that the RHSs $t = s$ of these implications, when viewed as reduction rule schemes $t \to s$, constitute a TRS of type 0. The condition $\bigwedge_{i=1}^{n} t_i = s_i$ will not be subject to restrictions. In particular it may contain variables not occurring in $t = s$.

In order to treat (*) as a *conditional reduction rule scheme*, some possibilities concerning the LHS $\bigwedge t_i = s_i$ arise, as expressed in the following definition. It will turn out (in 3.6) that only two of the four possibilities are sensible and interesting.

2.1. DEFINITION. (i) A *conditional TRS* Σ is a triple $\langle \mathscr{F}, \mathscr{V}, \mathbb{R} \rangle$, where $\mathscr{F}$ is a set of operators and $\mathscr{V}$ a set of variables and $\mathbb{R}$ is a set of *conditional reduction rule schemes* of the form

$$t_1 \square s_1 \wedge \cdots \wedge t_n \square s_n \Rightarrow t \rightarrow s.$$

Here $\square$ is $=$ (convertibility), $\downarrow$ (having a common reduct) or $\twoheadrightarrow$. Σ is called, respectively, to be of *type* I, II, or III.

(ii) If r is a conditional reduction rule scheme, r_u (the unconditional part of r) is the RHS $t \rightarrow s$ of r. Likewise $\mathbb{R}_u = \{r_u \mid r \in \mathbb{R}\}$ and $\Sigma_u = \langle \mathscr{F}, \mathscr{V}, \mathbb{R}_u \rangle$.

(iii) As before, $\mathrm{Ter}(\Sigma)$ is the set of terms of Σ, $\mathrm{Ter}^c(\Sigma)$ the set of closed terms and ρ denotes an instantiation.

(iv) An *unconditional normal form* of Σ is a normal form of Σ_u. (I.e. a term which cannot be unified with the LHS t of the RHS $t \rightarrow s$ of some $r \in \mathbb{R}$.)

We will mainly be interested in the following subclass of type III TRSs:

2.2. Definition. Let Σ be of type III where in every conditional rule scheme

$$t_1 \twoheadrightarrow n_1 \wedge \cdots \wedge t_k \twoheadrightarrow n_k \Rightarrow t \rightarrow s$$

the n_i $(i = 1, ..., k)$ are *closed unconditional normal forms*. Then Σ is called to be of type III_n.

2.3. Examples. (i) This example of an algebraic specification, modeled as a type III_n TRS, is given in [18]. The III_n-TRSs there considered, have conditional rule schemes of the form

$$\beta \twoheadrightarrow \mathit{true} \Rightarrow t \rightarrow s$$

where β is of boolean type; an important difference with the present paper is the hierarchical structure underlying the III_n-TRSs studied in [18]. (See Sect. 2.6 below.)

BOUNDED-STACK

sorts: *b-stack*, *entry*, *bool*, *int*
constants: 0, $M \in \mathit{int}$, $\mathit{true} \in \mathit{bool}$, $\varnothing \in$ *b-stack*, $\oplus \in \mathit{entry}$
functions: PUSH: *b-stack* $\times$ *entry* $\rightarrow$ *b-stack*
POP: *b-stack* $\rightarrow$ *b-stack*
TOP: *b-stack* $\rightarrow$ *entry*
$<$: $\mathit{int} \times \mathit{int} \rightarrow \mathit{bool}$ (less than)
$\#$: *b-stack* $\rightarrow$ *int* ($\#$: size)
S: $\mathit{int} \rightarrow \mathit{int}$ (S: successor)

axioms: $\#(\varnothing) \to 0$
$\#(\text{PUSH}(x, y)) \to S(\#(x))$
$M \to S(S(S(S(0))))$
$\text{POP}(\varnothing) \to \varnothing$
$\#(x) < M \twoheadrightarrow true \Rightarrow \text{POP}(\text{PUSH}(x, y)) \to x$
$\text{TOP}(\varnothing) \to \oplus$
$\#(x) < M \twoheadrightarrow true \Rightarrow \text{TOP}(\text{PUSH}(x, y)) \to y$

(ii) The following example is included merely for illustrative purposes. "Trivial combinatory logic," TCL, has the same operators I, K, S as CL in 1.2, and has conditional rule schemes:

$$a \twoheadrightarrow I \wedge b \twoheadrightarrow I \wedge c \twoheadrightarrow I \Rightarrow Sabc \to ac(bc)$$

$$a \twoheadrightarrow I \wedge b \twoheadrightarrow I \qquad \Rightarrow Kab \to a$$

$$a \twoheadrightarrow I \qquad \Rightarrow Ia \quad \to a$$

TCL is a type III_n TRS.

(iii) CL + the conditional rule scheme $x \downarrow y \Rightarrow D(x, y) \to E$ is a type II TRS.

2.4. *Generating the rules from the Conditional Rule Schemes*

If $r = \bigwedge_{i=1}^{k} t_i \twoheadrightarrow n_i \Rightarrow t \to s$ is a type III_n conditional rule scheme and ρ is an instantiation, then

$$\rho(r) = \bigwedge_{i=1}^{k} \rho(t_i) \twoheadrightarrow n_i \Rightarrow \rho(t) \to \rho(s)$$

is called a *conditional closed rule*. The word "closed" will sometimes be dropped; but the presence of conditions will always explicitly be mentioned. So a *rule* has the form $\rho(t) \to \rho(s)$, without conditions. The rules $\rho(t) \to \rho(s)$ which give rise to the *reduction steps* $C[\rho(t)] \to C[\rho(s)]$, are generated from $\mathbb{R}$, the set of conditional reduction rule schemes, as follows.

First we recall the notation $\bar{\mathbb{R}}$, for the set of closed instances of the conditional reduction rule schemes in $\mathbb{R}$, and $\mathscr{R}^0$ for the contextual, transitive reflextive closure of a binary relation $\mathscr{R}$ on $\text{Ter}^c(\Sigma)$ (a set of rules). In order to bring out the "least fixed point" aspect of the reduction $\to$ that is determined by $\mathbb{R}$, we define

2.4.1. DEFINITION (Application of sets of conditional rules). (i) Let $\mathscr{X}$ be a set of closed conditional rules $\bigwedge t_i \twoheadrightarrow n_i \Rightarrow t \to s$ and let $\mathscr{Y}$ be a set of closed rules $t_j \to s_j$ $(j \in I)$. Then $\mathscr{X}(\mathscr{Y})$ ("$\mathscr{X}$ applied to $\mathscr{Y}$") is the following set of closed rules:

$t \to s \in \mathscr{X}(\mathscr{Y}) \Leftrightarrow t \to s \in \mathscr{Y}$, or: there is a conditional rule $\bigwedge_{i<k} t_i \twoheadrightarrow n_i \Rightarrow t \to s$ in $\mathscr{X}$ such that $t_i \twoheadrightarrow n_i \in \mathscr{Y}^0$ for all $i < k$.

Notation. $\mathscr{X}^2(\mathscr{Y}) = \mathscr{X}(\mathscr{X}(\mathscr{Y}))$, etc.

(ii) Now let $\Sigma = \langle \mathscr{F}, \mathscr{V}, \mathbb{R} \rangle$ be a TRS of type III_n. Then $\mathscr{R}(\Sigma)$ is the set of rules of Σ, and we define

$$\mathscr{R}(\Sigma) = \bigcup_{n \in \omega} \bar{\mathbb{R}}^n(\varnothing).$$

(iii) Now the reduction relation $\to$ of Σ is $\mathscr{R}(\Sigma)^m$ (the contextual closure of $\mathscr{R}(\Sigma)$) and $\twoheadrightarrow$ is $\mathscr{R}(\Sigma)^{m*}$ $(= \mathscr{R}(\Sigma)^0)$.

(iv) We will define the *intermediate reductions* $\to_k$ $(k \in \omega)$:

$$\xrightarrow[k]{} = \Big(\bigcup_{n \leqslant k} \mathbb{R}^n(\varnothing) \Big)^m.$$

(So $\to_0 = \varnothing^m = \varnothing$ and $\twoheadrightarrow_0 = \varnothing^{m*} = \varnothing^* = \equiv$.)

(v) $\mathrm{Red}(\Sigma)$ is the set of *redexes*, i.e., the LHSs of elements of $\mathscr{R}(\Sigma)$. $\mathrm{NF}(\Sigma)$ is the set of of *normal forms*, i.e., terms not containing a redex.

2.4.1.1. *Remark.* (i) Note that $\to = \bigcup_{k \in \omega} \to_k$.

(ii) Definition 2.4.1 is given for type $\mathrm{III}_{(n)}$ conditional rule schemes, but it is obvious how to adapt the definition to the case of types I, II.

2.4.1.2. EXAMPLE. Consider TCL as in Example 2.3(ii). Then, e.g., $SIII \twoheadrightarrow I$, $S(SIII)\, II \twoheadrightarrow I$. However $SSII$ is a normal form, albeit not an unconditional one.

2.5. *Embedding Conditional TRSs in Unconditional Ones*

By introducing some more operators in a conditional TRS of type II or III, we can eliminate the conditions. That is, the conditional TRSs can be embedded in unconditional ones. We will not explore the more formal aspects of this embedding, but use it as a heuristic tool to construct the counterexamples to the CR-property for some type II and type III TRSs in the next section, and moreover we will use the embedding in order to state a natural criterion for decidability of the set of normal forms in a type III_n TRS Σ, in Section 4.

2.5.1. DEFINITION. Let $\Sigma = \langle \mathscr{F}, \mathscr{V}, \mathbb{R} = \{ r_i \mid i \in J \} \rangle$ be a TRS of type III.

(i) To each conditional rule scheme

$$r_i : \bigwedge_{j=1}^{k} t_j \twoheadrightarrow s_j \Rightarrow t \to s$$

we associate the pair of rule schemes r_i', r_i'' $(i \in J)$:

$$r_i' : t \to \delta_i\,(t_1, \ldots, t_k)\, s$$

$$r_i'' : \delta_i\,(s_1, \ldots, s_k) \to I.$$

(ii) $\Sigma_\delta = \langle \mathscr{F}_\delta, \mathscr{V}, \mathbb{R}_\delta \rangle$, where

$$\mathscr{F}_\delta = \mathscr{F} \cup \{I\} \cup \{\delta_i \mid i \in J\}$$
$$\mathbb{R}_\delta = \{r_i', r_i'' \mid i \in J\} \cup \{Ix \to x\}.$$

2.5.2. DEFINITION. Let $\Sigma = \langle \mathscr{F}, \mathscr{V}, \mathbb{R} = \{r_i \mid i \in J\} \rangle$ be a TRS of type II:

(i) To each conditional rule scheme

$$r_i \colon \bigwedge_{j=1}^{k} t_j \downarrow s_j \Rightarrow t \to s$$

we associate the pair of rule schemes r_i', r_i'' $(i \in J)$:

$$r_i' \colon t \to \delta_i(t_1, s_1, t_2, s_2, \ldots, t_k, s_k)\, s$$
$$r_i'' \colon \delta_i(x_1, x_1, x_2, x_2, \ldots, x_k, x_k) \to I.$$

(ii) Σ_δ is defined analogous to Definition 2.5.1.

To understand the next proposition we recall our basic assumption that Σ_u, the unconditional part of Σ, is a TRS of type 0.

2.5.3. PROPOSITION. (i) *Let Σ be of type* III_n. *Then Σ_δ is of type* 0.

(ii) *Let Σ be of type* II. *Then Σ_δ is of type* 0′ (*but not type* 0″).

Proof. Obvious. ∎

2.5.3.1. *Remark.* If Σ is of type III, Σ_δ may be ambiguous as well as non-left-linear.

2.5.4. PROPOSITION. *Let Σ be of type* III_n. *Then for all $t, s \in \mathrm{Ter}(\Sigma)$:*

$$\Sigma \models t \twoheadrightarrow s \Rightarrow \Sigma_\delta \models t \twoheadrightarrow s.$$

Proof. A routine induction on n (in $\twoheadrightarrow_n$); each Σ-reduction step can be simulated in Σ_δ, by construction. ∎

2.5.4.1. *Remark.* The reverse implication $(\Leftarrow)$ in Proposition 2.5.4 holds also, but since we have no need for it, we will omit a proof.

2.6. *Hierarchical Conditional TRSs*

In [18] an interesting class of III_n-TRSs is introduced and analyzed, namely conditional TRSs with a hierarchical structure. In order to define these hierarchically structured TRSs, first the following definition.

2.6.1. DEFINITION. (i) Let $\mathbb{R}$ be a set of conditional rule schemes, and $T \subseteq$

$\mathrm{Ter}^c(\Sigma)$ some set of terms. Then $\mathbb{R}^T$ ($\subseteq \bar{\mathbb{R}}$) is the set of all conditional rules obtained by instantiations $\rho: \mathscr{V} \to T$.

(ii) If $\Sigma = \langle \mathscr{F}, \mathscr{V}, \mathbb{R} \rangle$ and $\Sigma' = \langle \mathscr{F}', \mathscr{V}, \mathbb{R}' \rangle$ are TRSs, then $\Sigma \subseteq \Sigma' \Leftrightarrow \mathscr{F} \subseteq \mathscr{F}'$ and $\mathbb{R} \subseteq \mathbb{R}'$. Now Pletat *et al.* consider in [18] TRSs obtained as follows: Given is a finite chain $\Sigma_0 \subseteq \Sigma_1 \subseteq \cdots \subseteq \Sigma_n$, where $\Sigma_i = \langle \mathscr{F}_i, \mathscr{V}, \mathbb{R}_i \rangle$, $i \leqslant n$, $\mathbb{R}_0$ contains only unconditional rule schemes, R_{i+1} $(i < n)$ contains conditional rule schemes $\bigwedge t_j \twoheadrightarrow n_j \Rightarrow t \to s$ of type III_n such that the conditions $t_j \twoheadrightarrow n_j$ contain only terms $\in \mathrm{Ter}(\Sigma_i)$. (In fact the Σ_i $(i \leqslant n)$ in the definition of [18] are subject to signature restrictions; this does not seem essential, however.)

Furthermore, let Σ be Σ_n; then the set of closed rules of Σ, $R_h(\Sigma)$, is defined by the following inductive definition. (Cf. Definition 2.4.1; we write $R_h(\Sigma)$ instead of $R(\Sigma)$ here to denote that the hierarchy has to be taken into account.) Let T_i abbreviate $\mathrm{Ter}^c(\Sigma_i)$, $i = 0,..., n$:

$$R_h(\Sigma_0) = \mathbb{R}_0^{T_0}$$

$$R_h(\Sigma_{i+1}) = R_h(\Sigma_i) \cup \mathbb{R}_{i+1}^{T_i}(R_h(\Sigma_i)).$$

In order to have the CR property, [18] requires the property of "forward-preserving":

$$A \in T_i \text{ and } A \to B \in R_h(\Sigma_{i+1}) \Rightarrow A \to B \in R_h(\Sigma_i),$$

for all $i < n$. This property is implied by a syntactic requirement, viz. if $\bigwedge t_j \twoheadrightarrow n_j \Rightarrow t \to s$ is a conditional rule scheme in $\mathbb{R}_{i+1}$, then t contains a "new" operator $\in \mathscr{F}_{i+1} - \mathscr{F}_i$.

We note that the hierarchical approach does not yield always the same congruence on the set of terms as our definition. Namely: let $\mathscr{A}$ be an algebraic specification with conditional equations. Suppose to $\mathscr{A}$ we can associate a type III_n TRS $\Sigma_{\mathscr{A}}$, as in Example 2.3(i) ("BOUNDED STACK") which was taken from [18]. Then the reduction $\to$ which we have constructed as a "least fixed point," yields the same congruence as the initial algebra semantics of $\mathscr{A}$. We will not give the routine proof of this fact here.

However, when $\mathscr{A}$ is "partitioned" so as to obtain a hierarchical TRS $\Sigma_{\mathscr{A}}$, the reduction relation given by $R_h(\Sigma_{\mathscr{A}})$ may yield a congruence which is strictly coarser than the congruence of the initial algebra semantics. A simple example to show this is:

2.6.2. EXAMPLE. $\Sigma_0 = \langle \{P, \mathbb{Q}, 0\}, \mathscr{V}, \{P(\mathbb{Q}x) \to 0\} \rangle$,

$$\Sigma_1 = \langle \{P, \mathbb{Q}, 0, A, B, C\}, \mathscr{V}, \{P(\mathbb{Q}x) \to 0, C \to C, P(x) \twoheadrightarrow 0 \Rightarrow A(x) \to B\} \rangle$$

Now the chain $\Sigma_0 \subseteq \Sigma_1$ determines a hierarchical TRS in the sense of [18], which is "forward complete." According to our Definition 2.4.1, $R(\Sigma_1)$ contains $A(\mathbb{Q}C) \to B$, since also $P(\mathbb{Q}C) \to 0 \in R(\Sigma_1)$.

For the hierarchical TRS, $P(QC) \to 0 \notin R_h(\Sigma_0)$, since $C \notin \mathrm{Ter}^c(\Sigma_0)$. Hence $A(\mathbb{Q}C) \to B \notin R_h(\Sigma_1)$.

Probably it will be possible to extend the definition of hierarchical TRS in a simple way so as to obtain coincidence of the congruence thus determined and the congruence of the initial algebra semantics.

3. Confluence

Let us for the moment consider conditional TRSs, where the condition $\mathbb{Q}$ in a conditional reduction rule scheme

$$\mathbb{Q}(\mathbf{x}, \mathbf{y}) \Rightarrow t(\mathbf{x}) \to s(\mathbf{x})$$

is an arbitrary predicate. Here the variables $\mathbf{y}$ do not occur in the RHS of the implication. (Note that the intended meaning of the quantification of the variables $\mathbf{x}$, $\mathbf{y}$ is as follows:

$$\forall \mathbf{x}, \mathbf{y}[\mathbb{Q}(\mathbf{x}, \mathbf{y}) \Rightarrow t(\mathbf{x}) \to s(\mathbf{x})]$$

which is by predicate logic equivalent to $\forall \mathbf{x}[(\exists \mathbf{y}\ \mathbb{Q}(\mathbf{x}, \mathbf{y}) \Rightarrow t(\mathbf{x}) \to s(\mathbf{x})]$.)

Let Σ be a conditional TRS, where the conditional rule schemes have the form $\mathbb{Q}(\mathbf{x}, \mathbf{y}) \Rightarrow t(\mathbf{x}) \to s(\mathbf{x})$, and such that the unconditional part Σ_u is of type 0. Note that if ρ is an instantiation such that $\mathbb{Q}(\rho\mathbf{x}, \rho\mathbf{y})$ holds (whence $A \equiv \rho(t(\mathbf{x})) \to \rho(s(\mathbf{x})) \equiv B$ is a rule of Σ) and $C \subseteq A$ is a proper subredex, then because Σ_u is of type 0, $C \subseteq \rho(x_i)$ for some $x_i \in \mathbf{x}$ $(= x_1, \ldots, x_n)$.

Now suppose that we have two diverging reduction steps as in Fig. 6.

Then the construction of the corresponding elementary diagram needs the validity of the condition

$$\mathbb{Q}(\rho(x_1), \ldots, \rho(x_i)', \ldots, \rho(x_n), \rho(\mathbf{y})),$$

where $\rho(x_i)'$ results from $\rho(x_i)$ by contracting C.

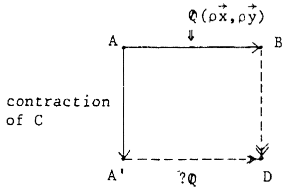

FIGURE 6

3.1. DEFINITION. If in the above situation for every ρ the validity of $\mathbb{Q}$ is preserved, then $\mathbb{Q}$ is called a *stable* condition.

3.2. THEOREM (O'Donnell [16]). *Let Σ be a conditional TRS with conditional rule schemes $\mathbb{Q}(\mathbf{x}, \mathbf{y}) \Rightarrow t(\mathbf{x}) \to s(\mathbf{x})$ such that Σ_u is of type* 0 *and all conditions $\mathbb{Q}$ are stable. Then Σ-reductions are confluent, and common reducts can be found by the canonical reduction diagram construction as in* 1.4.

Proof. The stability of the conditions ensures that elementary diagrams can be constructed, as if we were working in Σ_u. ∎

3.3. COROLLARY. *Type* I *reductions are confluent.*

Proof. Consider a type I conditional rule scheme:

$$t_1(\mathbf{x}, \mathbf{y}) = s_1(\mathbf{x}, \mathbf{y}) \wedge \cdots \wedge t_k(\mathbf{x}, \mathbf{y}) = s_k(x, y) \Rightarrow t(\mathbf{x}) \to s(\mathbf{x}).$$

Then the condition $\mathbb{Q}(\mathbf{x}, \mathbf{y})$ defined by the LHS of this implication is obviously stable, since if $t_i(\rho\mathbf{x}, \rho\mathbf{y}) = s_i(\rho\mathbf{x}, \rho\mathbf{y})$ then reduction in one of the $\rho(x_j)$ does not disturb the equality (as it is the transitive reflexive symmetric closure of reduction). ∎

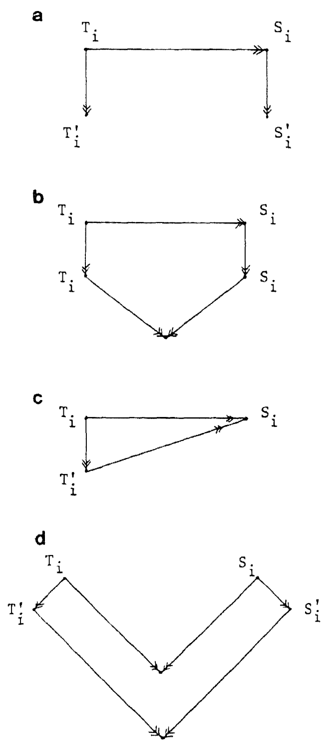

FIGURE 7

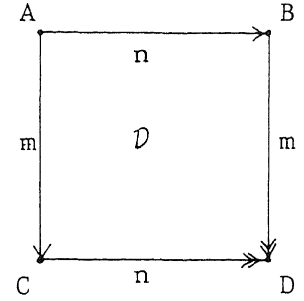

FIGURE 8

3.4. *Remark.* Intuitively, confluence for type III reductions is not plausible, since if

$$T_i \equiv t_i(\rho(\mathbf{x}, \mathbf{y})) \twoheadrightarrow s_i(\rho(\mathbf{x}, \mathbf{y})) \equiv S_i$$

(cf. the proof of Corollary 3.3) then reduction in one of the $\rho(x_j)$ may very well disturb the condition, as suggested in Fig. 7a. Then $T_i' \twoheadrightarrow S_i'$ will in general not be the case; even if CR would hold we have only the situation as in Fig. 7b. For III_n-reductions however, S_i is a closed normal form and hence we may hope to have stability (see Fig. 7c). Likewise, for II-reductions, stability is not a priori impossible; see Fig. 7d. Somewhat surprisingly, it will turn out that in the case of II-reductions, CR fails. First we establish the confluency of type III_n reductions.

3.5. THEOREM. *Let Σ be a type* III_n *TRS. Then Σ-reductions are confluent, and common reducts can be found by the canonical reduction diagram construction as in* 1.4.

Proof. We recall the definition of the intermediate reduction relations $\rightarrow_n$ $(n \in \omega)$ in Definition 2.4.1.

CLAIM. *Let $A \rightarrow_n B$ and $A \rightarrow_m C$. So $A \rightarrow B$ and $A \rightarrow C$. Let $\mathscr{D}$ be the elementary*

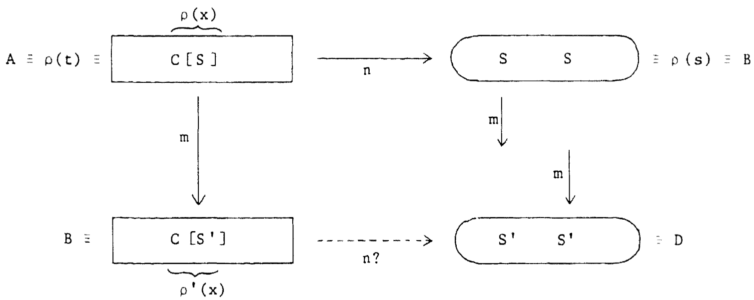

FIGURE 9

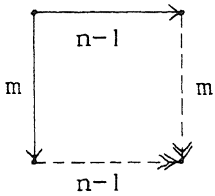

FIGURE 10

diagram determined by these two reduction steps. Then for the common reduct D (see fig. 8) *we have not only* $B \twoheadrightarrow D$ *and* $C \twoheadrightarrow D$, *but even* $B \twoheadrightarrow_m D$ *and* $C \twoheadrightarrow_n D$.

Clearly, the result in the theorem follows at once from the claim, since we already know that diagram constructions (as in 1.4) by repeatedly adjoining elementary diagrams, must terminate in a completed diagram.

Proof of the Claim. By induction to $n+m$. *Basis*: $n=m=0$. In this case the claim is vacuously true, since $\rightarrow_0$ is the empty relation.

Induction step. Suppose the claim is true for all n, m such that $n+m \leqslant k$. Consider n, m with $n+m=k+1$. Say $n>0$. The only interesting case is that where A is a redex, $A \equiv \rho(t)$, containing a proper subredex S which is contracted in the step $A \rightarrow_m C$ (see Fig. 9).

In the reduction $B \twoheadrightarrow D$, where copies of S are contracted, there is no problem: $B \twoheadrightarrow_m D$.

The question is, however, whether the step $C \rightarrow D$ is an n-step. Let the step $A \rightarrow B$ be generated by the conditional rule scheme $\bigwedge\!\!\!\bigwedge_{i \leqslant k} t_i \twoheadrightarrow n_i \Rightarrow t \rightarrow s$, via instantiation ρ. This means, by definition of $\rightarrow_n$, that $\rho(t_i) \twoheadrightarrow_{n-1} n_i$ for $i<k$. Because Σ_u is of the type 0, we have $S \subseteq \rho(x)$ for some x in t. Say $\rho(x) \equiv \mathrm{C}[S]$ for some context C[ ].

We have to prove that also $\rho'(t_i) \twoheadrightarrow_{n-1} n_i$ for $i<k$, where $\rho'(x) \equiv \mathrm{C}[S']$, S' is the contractum of S, and $\rho'(y) \equiv \rho(y)$ for $y \neq x$. For, then $C \equiv \rho'(t) \rightarrow_n D$ will be a consequence.

Now the induction hypothesis states that we have (see Fig. 10) (i.e., the claim holds for $n-1, m$). Say t_i contains three occurrences of x: $t_i \equiv ...x...x...x...$ and let, as before, $\rho(x)$ be $\mathrm{C}[S]$. Then $\rho(t_i) \equiv$ ---C[S]---C[S]---C[S]---. Let $q' \equiv$ ---C[S']---C[S]---C[S]--- and $q'' \equiv$ ---C[S']---C[S']---C[S]---, and $\rho'(t_i) \equiv$ ---C[S']---C[S']---C[S']---. Now we can construct a diagram, e.g., as in Fig. 11. Hence $\rho'(t_i) \twoheadrightarrow_{n-1} n_i$ $(i<k)$. This proves the claim and thereby the theorem. ∎

We will now show that type II and type III reductions are not confluent.

3.6. EXAMPLE. Consider the type II TRS Σ, where

$$\mathbb{R} = \begin{cases} x \downarrow C(x) \Rightarrow C(x) \rightarrow E \\ B \rightarrow C(B). \end{cases}$$

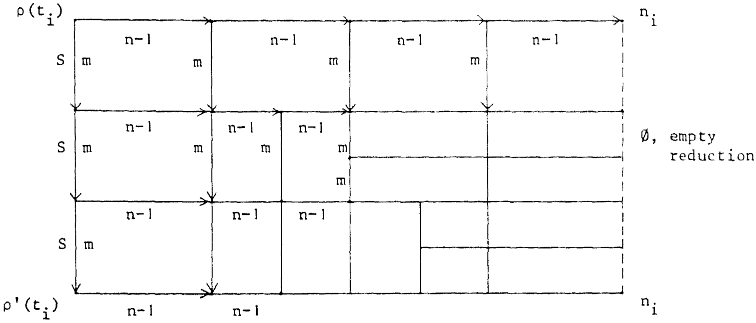

FIGURE 11

Then Σ_δ is a type 0′ TRS with

$$\mathbb{R}_\delta = \begin{cases} C(x) \to \delta(x, C(x))E \\ \delta(x, x) \to I \\ Ix \to x \\ B \to \mathrm{C}(B). \end{cases}$$

(Note that we use ranked and applicative notation simultaneously; cf. 1.2.) Cf. Example 1.5.1.0. As in Example 1.5.1.0, $\Sigma_\delta \not\models \mathrm{CR}$:

$$\begin{array}{l} B \to C(B) \to \delta(B, C(B))E \to \delta(\mathrm{C}(B), \mathrm{C}(B))E \to IE \to E \\ \qquad\quad \downarrow\!\!\!\downarrow \\ \qquad \mathrm{C}(E) \end{array}$$

and now $\mathrm{C}(E) \not\downarrow E$ as is easily seen. By analogy, we have also $\Sigma \not\models \mathrm{CR}$:

$$\begin{array}{l} B \to \quad C(B) \xrightarrow{\text{(since } B\downarrow C(B))} E \\ \qquad\qquad \downarrow \\ \qquad\quad \mathrm{C}(\mathrm{C}(B)) \\ \qquad\qquad \downarrow \\ \qquad\quad \mathrm{C}(E) \end{array}$$

and now $\mathrm{C}(E) \not\downarrow E$, as can easily be proved.

3.6.1. A variant of this counterexample, the type III TRS Σ' with

$$\mathbb{R} = \begin{cases} x \twoheadrightarrow C(x) \Rightarrow C(x) \to E \\ B \to C(B) \end{cases}$$

shows that type III reductions are in general not confluent.

3.6.2. EXAMPLE. Consider the type II TRS as in Example 2.3 (iii): $\Sigma = \text{CL} + \{x \downarrow y \Rightarrow Dxy \to E\}$. Then, intuitively, the CR-problem for Σ is the same as for $\Sigma_\delta = \text{CL} + \{Dxy \to \delta(x, y)E, \delta(x, x) \to I\}$. Again, it is intuitively clear that Σ_δ has the same CR-problem as $\Sigma'_\delta = \text{CL} + \{Dxy \to \delta'(x, y), \delta'(x, x) \to E\}$.

But this is nothing else than $\Sigma''_\delta = \text{CL} + \{Dxx \to E\}$ for which $\Sigma''_\delta \not\models \text{CR}$ by a counterexample analogous to the one in Example 1.5.1.0. (Cf. also Remark 1.5.2.2.) Hence $\Sigma \not\models \text{CR}$.

4. THE COMPLEXITY OF NORMAL FORMS

Given an unconditional TRS Σ, the set NF(Σ) of normal forms is clearly decidable. This is no longer true when Σ is of type I or III_n, in which cases the complexity of NF(Σ) can even be complete Π^0_1. (By the nonconfluence result of the last section we will no longer consider type II TRSs and type III TRSs in general.)

We will give some conditions for Σ in order to have a decidable set of normal forms, which is important if one wants to use terminating reduction strategies (see Sect. 5).

4.1. DEFINITION. Let Σ be a TRS (of type 0, I, III_n):

(i) Then the set of *normal forms of* Σ, NF(Σ), is the set of Σ-terms M such that $\neg\exists N,\ M \to N$ (i.e., admits no reduction step from M).

(ii) Let Σ_u be the unconditional TRS (so of type 0) associated with Σ. Then $\text{NF}(\Sigma_u) \subseteq NF(\Sigma)$ is called the set of *unconditional normal forms* of Σ.

(iii) Let Σ have the conditional rule schemes $r_1, \ldots, r_n$. Then $M \in \text{Ter}(\Sigma)$ is a r_i*-preredex* if M is a$(r_i)_u$-redex of Σ_u. (Recall that $(r_i)_u$ is the unconditional part of r_i.)

In the case of III_n-TRSs, which are our main interest, the normal forms are naturally partitioned in a hierarchy, as follows.

4.2. DEFINITION. Let Σ be a III_n-TRS:

(i) By induction on n we will define the set $\text{NF}_n(\Sigma) \subseteq \text{NF}(\Sigma)$ of *normal forms of order n*.

Basis. $\text{NF}_0(\Sigma) = \text{NF}(\Sigma_u)$, the set of unconditional normal forms.

Induction step. Suppose the set of normal forms of order n, $\mathrm{NF}_n(\Sigma)$, is defined. Then $\mathrm{NF}_{n+1}(\Sigma)$ is defined by:

$M \in NF_{n+1}(\Sigma)$ iff whenever $M' \subseteq M$ is an r-preredex (where r is a conditional rule scheme of Σ and r is $t_1 \twoheadrightarrow n_1 \wedge \cdots \wedge t_k \twoheadrightarrow n_k \Rightarrow t \to s$, so M' is an instance of t, say $M' \equiv \rho(t)$), then for some $j \in \{1,..., k\}$:

$$\exists l \leqslant n, \quad \exists N \in \mathrm{NF}_l(\Sigma), \quad \rho(t_j) \twoheadrightarrow N \quad \text{and} \quad N \not\equiv n_j.$$

We will call a normal form of order n also a *n-normal form.*

(ii) $\mathrm{NF}_f(\Sigma)$, the set of normal forms of finite order, is $\bigcup_{n \in \omega} \mathrm{NF}_n(\Sigma)$.

4.2.1. PROPOSITION. (i) $\mathrm{NF}_0 \subseteq \mathrm{NF}_1 \subseteq \mathrm{NF}_2 \subseteq \cdots$.

(ii) $\mathrm{NF}_f \subseteq \mathrm{NF}$.

Proof. (i) Obvious; (ii) Follows by a simple induction from the CR property for III_n TRSs (Theorem 3.5), noting that CR implies unicity of normal forms. ∎

So we have a "spectrum" of irreducibility as in Fig. 12.

4.3. EXAMPLE. Consider TCL as in Example 2.3(ii). Then SII is a 0-normal form, $\Omega \equiv SII\,(SII)$ is a 1-normal form, $S\,\Omega\,\Omega\,\Omega$ is a 2-normal form. In fact, every non-reducible term will be in this case a normal form of finite order (by Proposition 4.6 below).

4.4. PROPOSITION. *Let $\Sigma = \langle \mathscr{F}, \mathscr{V}, \mathbb{R} \rangle$ be of type* III_n. *Suppose $\mathbb{R}$ is finite. Then*:

(i) *The set* $\mathrm{NF}_f(\Sigma)$ *of normal forms of finite order is semi-decidable.*

(ii) *The set* $\mathrm{NF}(\Sigma)$ *of normal forms may be undecidable.*

Proof. (i) is apparent from the definition. (ii). Consider the TRS CL, as in 1.2. It is well known that the natural numbers can be represented by CL-terms **n**, which are in normal form; furthermore, there exists a CL-term E, also in normal form, which acts as an enumerator in the sense that, if $\ulcorner\ \urcorner$: Ter(CL) $\to \mathbb{N}$ is a recursive coding of CL-terms:

$$E\ulcorner M \urcorner \twoheadrightarrow M$$

for all $M \in$ Ter (CL). For a proof, see [1, Theorem 8.1.6].

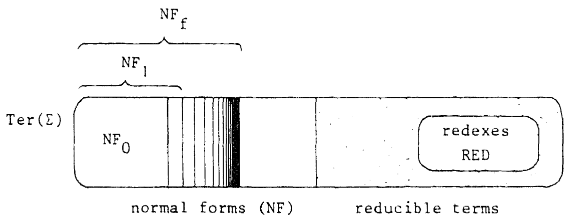

FIGURE 12

Now consider $\Sigma = \text{CL}$ extended by a new operator T and the conditional rule

$$Ex \twoheadrightarrow \mathbf{0} \Rightarrow Tx \rightarrow \mathbf{1}.$$

Note that the Σ-reduction $\rightarrow$, thus obtained, satisfies

$$Ex \twoheadrightarrow \mathbf{0} \Leftrightarrow Tx \rightarrow \mathbf{1}.$$

Hence, if $\text{NF}(\Sigma)$ were decidable, the set

$$\{M \in \text{Ter(CL)} \mid EM \twoheadrightarrow \mathbf{0}\}$$

and in particular

$$\{\mathbf{n} \in \text{Ter(CL)} \mid E\mathbf{n} \twoheadrightarrow \mathbf{0}\}$$

would be decidable. Since $\Sigma \models \text{CR}$ (Theorem 3.5) and noting that, hence, $E\mathbf{n} \twoheadrightarrow M$ and $E\mathbf{n} \twoheadrightarrow \mathbf{0}$ implies $M \twoheadrightarrow \mathbf{0}$, this would mean that

$$\{M \in \text{Ter(CL)} \mid M \twoheadrightarrow \mathbf{0}\}$$

is a decidable set, which is not true. (This follows, e.g., from a theorem of Scott, see [1, Theorem 6.6.2], as follows:
If $\varnothing \subsetneq \mathscr{X} \subsetneq \text{Ter(CL)}$ *and* $\mathscr{X}$ *is closed under equality, then* $\mathscr{X}$ *is not recursive.*)
So $\text{NF}(\Sigma)$ is not decidable. ∎

4.4.1. *Remark.* If $\text{NF}(\Sigma)$ is not decidable, it is clearly also not semi-decidable, since the complement $\text{Ter}(\Sigma) - \text{NF}(\Sigma)$ is semi-decidable. Being the complement of a semi-decidable set (i.e., of complexity Σ^0_1), $\text{NF}(\Sigma)$ has always compexity Π^0_1. For Σ as in the proof of Proposition 4.4 (ii), it is not hard to show that $\text{NF}(\Sigma)$ is complete Π^0_1.

Next we will state some conditions for III_n-TRSs which ensure the decidability of the set of normal forms.

4.5. Definition. (i) Let Σ be a III_n-TRS. Then Σ "*has subterm conditions*" iff for every instance of a conditional rule scheme

$$\rho(t_1) \twoheadrightarrow n_1 \wedge \cdots \wedge \rho(t_k) \twoheadrightarrow n_k \Rightarrow \rho(t) \rightarrow \rho(s)$$

we have

$\rho(t_i) \subsetneq \rho(t)$ (i.e., $\rho(t_i)$ is a proper subterm of $\rho(t)$) for all $i = 1, \ldots, k$.

(ii) As a special case of (i), we say that Σ "*has variable conditions*" iff every conditional rule scheme is of the form

$$x_1 \twoheadrightarrow n_1 \wedge \cdots \wedge x_k \twoheadrightarrow n_k \Rightarrow t \rightarrow s$$

where $x_1, \ldots, x_k$ are variables occurring in t.

4.6. PROPOSITION. *If Σ is a* III_n-TRS *having subterm conditions, then*:

(i) $\mathrm{NF}(\Sigma) = \mathrm{NF}_f(\Sigma)$

(ii) $\mathrm{NF}(\Sigma)$ *is decidable.*

Proof. (i) Let M be a term which is not reducible, and suppose that M is not a normal form of finite order. Choose M minimal so, w.r.t. $\subseteq$. Hence all proper subterms of M are normal forms of finite order. Let m be the maximum of their orders. Then clearly M is a normal form of order $m+1$, since Σ has subterm conditions.

(ii) The set of reducible terms is semi-decidable (just generate all possible finite reductions, as in Definition 2.4.1). By Proposition 4.4(i) and (i) of this proposition, its complement NF is also semi-decidable. Hence both the set of reducible terms and NF are decidable. ∎

4.6.1. EXAMPLE. TCL, in Example 2.3 (ii), has variable conditions. Hence NF is decidable.

4.7. DEFINITION. Let Σ be a TRS:

(i) Then $\Sigma \models \mathrm{SN}$ ("Σ has the strong normalization property") iff there are no infinite Σ-reductions. Equivalently, iff every Σ-reduction terminates eventually (in $\mathrm{NF}(\Sigma)$).

(ii) $\Sigma \models \mathrm{WN}$ ("weak normalization") iff every $M \in \mathrm{Ter}(\Sigma)$ has a normal form, i.e., there exists an $\mathscr{R} = M \to \cdots \to N$ with $N \in \mathrm{NF}(\Sigma)$.

4.8. DEFINITION. Let Σ be a III_n-TRS. Then:

(i) $\Sigma \models \mathrm{SN}_0$ iff every reduction terminates eventually in a 0-normal form;

(ii) $\Sigma \models \mathrm{SN}_f$ iff every reduction terminates eventually in a normal form of finite order.

4.9. THEOREM. (*Criteria for NF-decidability in* III_n*-reductions*). *Let Σ be a* III_n-TRS. *Then the following implications hold.* (*see Fig.*13).

Proof. (v) ⇒ (vii) ⇒ (viii) is Proposition 4.6. (iii) ⇒ (vi), (i) ⇒ (iv) ⇒ (vi), and (iv) ⇒ (vii) follow trivially from the definitions. To prove (ii) ⇒ (iv), assume $\Sigma_\delta \models \mathrm{SN}$. By Proposition 2.5.4, $\Sigma \models \mathrm{SN}$. Hence it suffices to prove $\mathrm{NF}_f(\Sigma) = \mathrm{NF}(\Sigma)$. For a proof by contradiction, suppose there is a normal form M without finite order. Say $M \equiv C[\rho(t)]$ for some conditional rule scheme $t_1 \twoheadrightarrow n_1 \wedge \cdots \wedge t_k \twoheadrightarrow n_k \Rightarrow t \to s$ and some context C[ ]. By SN, all $\rho(t_i)$ $(i = 1,\dots, k)$ have a normal form n_i'. One of the n_i' must be wrong ($n_i' \not\equiv n_i$) and without finite order. Say n_{i_0}' is such a wrong normal form without finite order. Write $M' \equiv n_{i_0}'$.

Since M' is a normal form without finite order, the same reasoning as for M

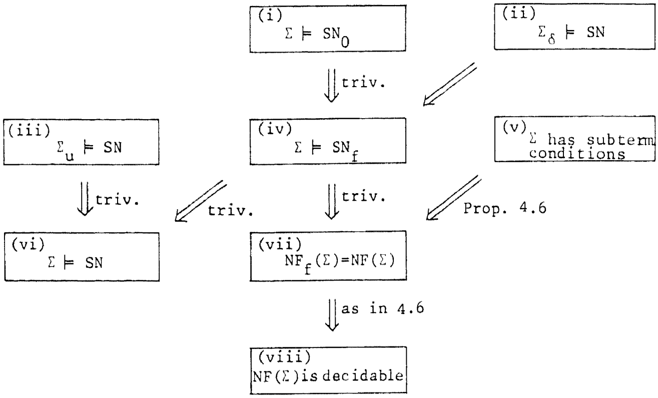

FIGURE 13

applies to M'. Continuing in this way we find an infinite sequence M, M', M'',.... This sequence is reflected in an infinite reduction in Σ_δ as follows (here we use Proposition 2.5.4 which says that reductions in Σ can be simulated in Σ_δ):

$$M \equiv C[\rho(t)] \to C[\delta(\rho(t_1),..., \rho(t_{i_0}),..., \rho(t_k))\ \rho(s)]$$

$$\Big\downarrow \text{ Proposition 2.5.4}$$

$$C[\delta(\rho(t_1),..., M',..., \rho(t_k))\ \rho(s)]$$

and so on. ∎

4.9.1. *Remark.* Most of the valid implications between (i),..., (viii) are displayed in the diagram of implications in Theorem 4.9. Several of the non-implications follow by considering the next example. A positive answer to the following question would yield a useful criterion for NF-decidability: does (iii) $\Rightarrow$ (viii) hold? ((iii) $\not\Rightarrow$ (vii) as the next example shows.)

4.10. EXAMPLE. (i) Let Σ have as operators: A, B, C, D, E, F, 0, all of arity 0, and conditional rule schemes:

$$C \twoheadrightarrow 0 \Rightarrow A \to B$$

$$C \to D$$

$$F \twoheadrightarrow 0 \Rightarrow D \to E$$

$$F \to A.$$

Then $NF(\Sigma) = \{A, B, D, E, 0\}$ and $NF_f(\Sigma) = \{B, E, 0\}$. Since $\mathrm{NF}(\Sigma) \neq \mathrm{NF}_f(\Sigma)$, we must have $\Sigma_\delta \not\models$ SN. Indeed this is the case; Σ_δ has rule schemes:

$$A \to \delta CB$$
$$\delta 0 \to I$$
$$Ix \to x$$
$$D \to \delta' FE$$
$$\delta' 0 \to I$$
$$C \to D$$
$$F \to A$$

and now $A \to \delta CB \to \delta DB \to \delta(\delta' FE) B \to \delta(\delta' AE) B \to \cdots$ yields an infinite reduction.

(ii) Σ has as only scheme the conditional rule scheme

$$L(L(x)) \twoheadrightarrow 0 \Rightarrow L(x) \twoheadrightarrow 1.$$

Then $L(0)$ is a normal form without finite order. In fact,

$$\mathrm{Ter}(\Sigma) = \mathrm{NF}(\Sigma); \mathrm{NF}_f(\Sigma) = \{0, 1\}.$$

4.11. *Remark.* Also in the approach with hierarchical conditional TRSs (Sect. 2.6), the problem of decidability of the set of redexes, $\mathrm{RED}(\Sigma)$, and of the set of normal forms, $\mathrm{NF}(\Sigma)$, arises. (The example in the proof of Proposition 4.4(ii), where $\mathrm{NF}(\Sigma)$ was complete $\prod_1^0$, applies also in the hierarchical case.)

5. Termination

In this section we will mention some criteria, given in [14], for termination, i.e., properties implying $\Sigma \models \mathrm{SN}$, which hold for Σ of type 0 and which generalize to types I, III_n. The proofs are verbatim the same as those for type 0 in [14] and will not be repeated here.

We will suppose that some "oracle" is given telling us what the redexes of Σ are (i.e., the LHSs of the rules in $R(\Sigma)$ as defined in 2.4.1). Let $\mathrm{RED}(\Sigma)$ be the set of Σ-redexes. In this connection, let us mention the

Question. Are the following equivalent?

(i) $\mathrm{NF}(\Sigma)$ is decidable.

(ii) $\mathrm{RED}(\Sigma)$ is decidable.

(ii) $\Rightarrow$ (i) is trivial. Furthermore, it is easy to show that

$$\Sigma \models \mathrm{SN} \text{ and } \mathrm{NF}(\Sigma) \text{ decidable} \Rightarrow \mathrm{RED}(\Sigma) \text{ decidable}.$$

However, since we are concerned with termination criteria and, in the next section, with terminating reduction strategies, this concern would trivialize when SN is already assumed.)

5.1. DEFINITION. (i) A rule scheme $t \to s$ is *non-erasing* when t, s have the same variables (e.g., $Kxy \to x$ is an erasing rule scheme).

(ii) A type 0 TRS Σ is non-erasing when all its rule schemes are.

(iii) A type I or III_n TRS Σ is non-erasing when Σ_u is non-erasing.

Notation. $\Sigma \models \text{NE}$.

5.2. THEOREM. *Let* Σ *be of type* I *or* III_n. *Then*: $\Sigma \models \text{NE} \Rightarrow (\Sigma \models \text{WN} \Leftrightarrow \Sigma \models \text{SN})$. (*For* WN, SN *see Definition* 4.7.)

So in order to prove strong normalization for a non-erasing TRS of types I, III_n it is sufficient to prove weak normalization.

5.3. DEFINITION. Let Σ be of type I or III_n. $\Sigma \models \text{WIN}$ (weak innermost normalization) iff every Σ-term has a normal form which can be found by reducing innermost Σ-redexes.

5.4. THEOREM (O'Donnell [16]). *Let* Σ *be of types* I *or* III_n. *Then*

$$\Sigma \models \text{WIN} \Leftrightarrow \Sigma \models \text{SN}.$$

5.5. DEFINITION. Let Σ be of type I or III_n. $\Sigma \models \text{DR}$ (decreasing redexes) iff there is a map d: $\text{RED}(\Sigma) \to \mathbb{N}$, such that

(i) if R' is a residual of R in some reduction step, then $d(R) \geqslant d(R')$;

(ii) if R' is created by contraction of R in some reduction step, then $d(R) > d(R')$.

5.6. THEOREM. *Let* Σ *be of type* I *or* III_n. *Then*

$$\Sigma \models DR \Rightarrow \Sigma \models \text{SN}.$$

5.7. *Terminating Reduction Strategies*

Analogous to the previous section, also the main results about terminating reduction strategies for type 0 TRSs carry over to the case of I or III_n TRSs. In order to execute strategies, we assume again an oracle deciding for us whether a Σ_u-redex is also a Σ-redex.

For the definitions of the following strategies we refer to [14, 16, 18].

5.7.1. THEOREM. *Let Σ be a type* I *or* III_n *TRS. Then the following are terminating reduction strategies (i.e., find the normal form when it exists):*

(i) *the "full substitution" strategy (or "full computation" strategy)*

(ii) *the "parallel outermost" strategy.*

Proof. (i) As for the type 0 case, see [14].

(ii) As for the type 0 case, see [14]; or see the Appendix. ∎

6. POSSIBLE EXTENSIONS

In this section we will mention some directions in which the preceding results can be generalized, and a direction in which such a generalization fails.

6.1. *Disjunctions.* It is not hard to prove that also disjunctions may be allowed in the LHS of a type I or III_n conditional reduction rule scheme, while retaining the confluence results. E.g.,

$$x \twoheadrightarrow 0 \vee (x \twoheadrightarrow 1 \wedge y \twoheadrightarrow 0) \Rightarrow P(x, y) \rightarrow Q$$

is such a type III_n conditional rule scheme. The "effect" of this conditional rule scheme is the same as that of the pair of conditional rule schemes

$$r_0: x \twoheadrightarrow 0 \Rightarrow P(x, y) \rightarrow Q$$

$$r_1: x \twoheadrightarrow 1 \wedge y \twoheadrightarrow 0 \Rightarrow P(x, y) \rightarrow Q.$$

(If Σ contains such a pair r_0, r_1, where $(r_0)_u = (r_1)_u$, Σ_u will be ambiguous; but this ambiguity is entirely harmless.)

6.2. *Infinite disjunctions.* In the same way we may admit infinite disjunctions in the LHS of a type I or III_n conditional rule scheme. Thus we obtain rules like

$$\bigvee_{N \in \text{NF}(\Sigma_u)} x \twoheadrightarrow N \Rightarrow P(x) \rightarrow Q.$$

(If x has an unconditional normal form, then $P(x) \rightarrow Q$.)

6.3. *Bound Variables.* It is also possible to derive the preceding results (except the one about WIN, in Theorem 5.4) for CRSs as in [14], i.e., TRSs with bound variables, having reduction rule schemes like, e.g.,

$$(\lambda x.\, A(x))B \rightarrow A(B)$$

$$\mu x.\, A(x) \rightarrow A(\mu x.\, A(x))$$

$$C(\lambda x.\, M(x), \lambda y.\, N(y)) \rightarrow \lambda y.\, M(N(y))$$

In the Appendix we generalize a result of O'Donnell to this case.

6.4. *Ambiguous* TRSs. In [9] a confluence theorem is proved for (unconditional) TRSs that are left-linear, but may be ambiguous (i.e., have critical pairs, see [9]):

6.4.1. THEOREM (Huet [9]). *If T is a left-linear TRS and for every critical pair $\langle P, Q\rangle$ we have $P \mathrel{\#\!\!\twoheadrightarrow} Q$, then T is confluent.*

(Here $\#\!\!\twoheadrightarrow$ denotes parallel reduction at disjoint occurrences.) We remark that the confluence of TRSs as in Huet's theorem is immediately disturbed when conditions are added of types I, or III_n. The following TRS Σ provides a simple counterexample to the CR property:

$$\Sigma \begin{cases} P(Q(x)) \to P(R(x)) \\ Q(H(x)) \to R(x) \\ \quad S(x) \twoheadrightarrow 0 \Rightarrow R(x) \to R(H(x)) \\ \quad\; S(x) \to 1. \end{cases}$$

The only critical pair of Σ_u is $\langle A, B\rangle$ as in the diagram:

$$\begin{array}{ccc} P(Q(H(x))) & \longrightarrow & P(R(H(x))) \equiv B \\ \downarrow & & \vdots \\ A \equiv P(R(x)) & \overset{\#}{\longrightarrow} & P(R(H(x))) \equiv B \end{array}$$

Indeed $A \mathrel{\#\!\!\twoheadrightarrow} B$ in Σ_u, hence $\Sigma_u \models \text{CR}$ by Huet's theorem. However in Σ the terms A, B have no common reduct, since the condition $S(x) \twoheadrightarrow 0$ is never true.

7. APPENDIX: PARALLEL OUTERMOST AND LEFTMOST REDUCTIONS

In this Appendix we will give an account of O'Donnell's ingenious proof that parallel outermost reductions are terminating whenever possible, and likewise for leftmost reductions if an additional assumption is made. Our version of the proof will illustrate our terminology of reduction diagrams, which, we feel, exhibits the structure of the proof more clearly. Moreover, we will prove a strengthened version, applying also to the case of term rewriting systems with bound variables (e.g., a TRS containing λ-calculus). This answers a suggestion in O'Donnell [16 (Further Research, p. 102]), namely to generalize his Theorem 10 to "SRSs with pseudo-residual maps." In fact, our generalization goes further than that; it applies also to the class of "combinatory reduction systems" as in [14].

7.1. PROPOSITION. *Let $\mathscr{D}$ be an elementary reduction diagram as in Fig.* 14 *and let $R_i \subseteq M_i$ $(i = 0, 2, 3)$ be redexes such that $R_0 -\cdot-\cdot \to R_2 -\cdot-\cdot \to R_3$. (See*

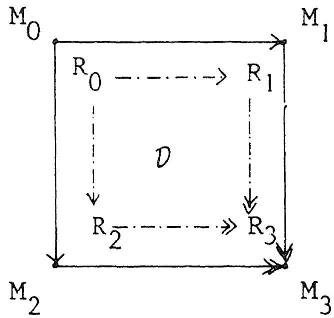

FIGURE 14

Definition 1.4.1.1.) *Then there is a unique redex* $R_1 \subseteq M_1$ *such that* $R_0 -\cdot-\cdot \rightarrow R_1 -\cdot-\cdot \rightarrow R_3$.

Proof. Routine. ∎

7.2. DEFINITION. Let π be a predicate on pairs of terms M, R such that $R \subseteq M$ and R is a redex. (If it is clear what M is meant, we will call R such that $\pi(M, R)$ a "π-redex.")

(i) π has property I if, in the situation of Proposition 7.1: ($\pi(M_0, R_0)$ and $\pi(M_2, R_2)$ and $\pi(M_3, R_3)) \Rightarrow \pi(M_1, R_1)$.

(ii) π has property II if in every reduction step $M \rightarrow^R M'$ such that $\neg\pi(M, R)$, every redex $S' \subseteq M'$ such that $\pi(M', S')$ has an ancestor redex $S \subseteq M$ with $\pi(M, S)$ ($\neg\pi$-steps cannot create new π-redexes).

7.3. PROPOSITION (*Separability of developments*). *Let* π *have property* II. *Then every development* $\mathcal{R} = M_0 \rightarrow \cdots \rightarrow M_n$ *can be separated into a "*π*-part" followed by a "*$\neg\pi$*-part"; i.e., there are reductions* $\mathcal{R}_\pi$: $M_0 \equiv N_0 \rightarrow^{R_0} \cdots \rightarrow^{R_{k-1}} N_k$ *such that* $\pi(N_i, R_i)$ $(i<k)$ *and* $\mathcal{R}_{\neg\pi}$: $N_k \rightarrow^{R_0} \cdots N_{k+l} \equiv M_n$ *such that* $\neg\pi(N_j, R_j)$ $(k \leqslant j < k+l)$. *Moreover,* $\mathcal{R}$ *is equivalent to* $\mathcal{R}_\pi * \mathcal{R}_{\neg\pi}$ ("$*$" *denotes concatenation*).

Proof. Let $\mathcal{R}$ be a development of some set $\mathbb{R}$ of redexes in M_0. Let these be characterized by underlining their head symbol. Contracting each step an arbitrary underlined π-redex, must lead to a term in which all remaining underlined redexes are $\neg\pi$-redexes. (This is so by the "Finite Developments" Lemma 1.4.3.)

Then we start contracting the underlined $\neg\pi$-redexes. By property II, this process will not create new underlined π-redexes. Also this $\neg\pi$-part of the development stops eventually.

The equivalence follows because all developments of the same $\mathbb{R}$ are equivalent (Proposition 1.4.3.1). ∎

7.3.1. *Remark.* For TRSs we do not need this proposition in the proof of Theorem 7.8. When bound varables are present, we do.

7.4. EXAMPLE. (i) *$\pi(M, R) \Leftrightarrow R$ is a redex.* Then properties I, II hold (I is Proposition 7.1 and II is vacuously true.)

(ii) *$\pi(M, R) \Leftrightarrow R$ is an outermost redex in M.* That property I holds can be seen as follows: consider the situation as in the hypothesis of Proposition 7.1, where moreover R_0, R_2, R_3 are outermost. Let S_i be the redex contracted in $M_0 \to M_i$, $i = 1, 2$. Suppose R_1 (as in the Proposition) is not outermost. This can only be the case if in M_1 a redex P is *created* which covers R_i. However, in $M_1 \twoheadrightarrow M_3$ redex R_1 becomes outermost again, which can only be the case if P is contracted. But this is not so since in $M_1 \twoheadrightarrow M_3$ residuals of S_2 are contracted (and P is not a residual of S_2, being created).

Property II is easily verified; it follows by what in [16] is called the "outer" property, which holds for every regular TRS.

(iii) *$\pi(m, R) \Leftrightarrow R$ is the leftmost redex of M.* Without additional assumptions, property II does not hold. Example (of [10]): $\Sigma = \{F(x, B) \to D,\ C \to C,\ A \to B\}$. Then the step $F(C, A) \to F(C, B)$ is a counterexample.

7.5. DEFINITION. (i) Let $\mathscr{R} = M_0 \to M_1 \to \cdots$ be a (finite or infinite) reduction. Let M_j be some fixed term in $\mathscr{R}$ ($j = 0, 1, 2,...$). Let $L_i \subseteq M_i$ for all $i \geqslant j$ as far as M_i is defined, such that $L_j - \cdot - \cdot > L_{k+1} - \cdot - \cdot > \cdots$. Then this sequence is called a *trace (of descendants) in $\mathscr{R}$.*

(ii) Let the L_i as in (i) be redexes, and suppose π is a predicate as in Definition 7.2. Then the trace $\mathscr{L}$ is a *π-trace* iff $\forall i \geqslant j,\ \pi(M_i, L_i)$.

(iii) Let $\mathscr{R}$ be a reduction and π be a predicate. Then $\mathscr{R}$ is *π-fair* iff $\mathscr{R}$ contains no infinite π-traces.

7.5.1. EXAMPLE. Let π be as in Example 7.4(i), (ii), (iii), respectively. Then π-fair reductions are called in [16]: *complete*, respectively *eventually outermost*, respectively *leftmost reductions.*

7.6. PROPOSITION. *Let π be a predicate as in Definition* 7.2 *having property* I. *Let $\mathscr{D}$ be an arbitrary reduction diagram as in the figure, where $R_i \subseteq M_i$ ($i = 0, 2, 3$) are redexes such that $R_0 - \cdot - \cdot \twoheadrightarrow R_2 - \cdot - \cdot \twoheadrightarrow R_3$ is a π-trace. Then the unique trace $R_0 - \cdot - \cdot \twoheadrightarrow R_1 - \cdot - \cdot \twoheadrightarrow R_3$ leading via M_1, is also a π-trace.* (*See Fig.* 15.)

Proof. Consider the completed reduction diagram $\mathscr{D}$. Then the trace of descendants $R_0 - \cdot - \cdot \twoheadrightarrow R_2 - \cdot - \cdot \twoheadrightarrow R_3$ can be pushed upwards in stages, each stage one elementary diagram further. Result: a trace $R_0 - \cdot - \cdot \twoheadrightarrow R_1 - \cdot - \cdot \twoheadrightarrow R_3$. (See Fig. 16.) Moreover, since the initial trace was a π-trace the resulting trace is by property I also a π-trace. ∎

7.7. PROPOSITION (*π-traceability is invariant under equivalence of reductions*). *Let π have property* I. *Let $\mathscr{R}$ and $\mathscr{R}'$ be equivalent finite reductions from M_0 to M_n. Let*

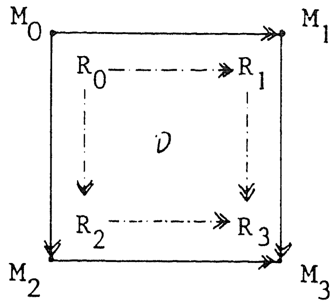

FIGURE 15

$S \subseteq M_0$, $S' \subseteq M_n$ be redexes such that there is a π-trace S—·—· $\twoheadrightarrow S'$ via $\mathscr{R}$. Then there is also such a π-trace via $\mathscr{R}'$, which is moreover unique.

Proof. See Fig. 17. By Proposition 7.6, the π-trace from S to S' via M_n as displayed in Fig. 17 can be pushed down to a π-trace via M'_k. Since the right and bottom side of $\mathscr{D}(\mathscr{R}, \mathscr{R}')$ consist of trivial steps, the result follows. ∎

7.8. THEOREM (O'Donnell [16]). *Let π be a predicate satisfying properties* I, II *of Definition* 7.2. *Then the class of π-fair reductions is closed under projections.*

Proof. See Fig. 18. Let $\mathscr{R} = M_0 \to M_1 \to \cdots$ and $S \subseteq M_0$ be a redex. Let $\mathscr{R}/\{S\}$ be a projection of $\mathscr{R}$. Suppose $\mathscr{R}$ is π-fair.

Let $M_k \twoheadrightarrow A_k \twoheadrightarrow N_k$ be a rearrangement of $M_0 \twoheadrightarrow M_k/\{S\}$ into a π-part followed by a $\neg\pi$-part, according to Proposition 7.3. Since the rearranged reduction is equivalent with the original one, the lower side of $\mathscr{D}(M_k \twoheadrightarrow A_k \twoheadrightarrow N_k, M_k \to M_{k+1})$ (the "curved" reduction $N_k \twoheadrightarrow N_{k+1}$ in Fig. 18) is equivalent to the original ("straight") reduction $N_k \twoheadrightarrow N_{k+1}$. By Proposition 7.7, the trace R_k—·—· $\twoheadrightarrow R_{k+1}$ via the curved reduction $N_k \twoheadrightarrow N_{k+1}$ is also π-fair.

Next we rearrange $M_{k+1} \twoheadrightarrow B_{k+1}$, given as $M_k \twoheadrightarrow A_k/M_k \to M_{k+1}$, into a

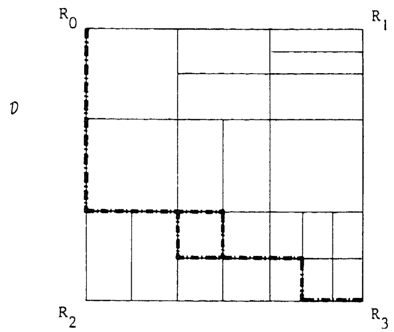

FIGURE 16

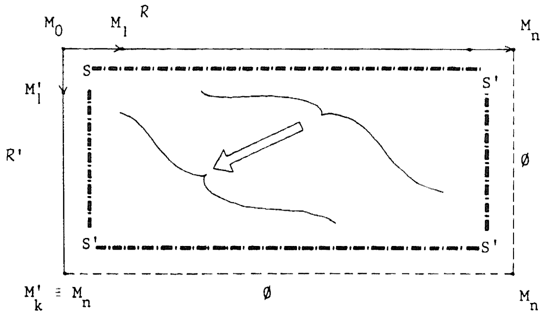

FIGURE 17

π-part followed by a $\neg\pi$-part. Iteration of this procedure leads to the "staircase" $A_k - B_{k+1} - A_{k+1} - B_{k+2} - \cdots$. (See Fig. 18.) This staircase reaches $\mathcal{R}$ after finitely many steps, for otherwise $\mathcal{R}$ would contain an infinite trace of descendants of S with property π, in contradiction with the π-fairness of $\mathcal{R}$.

Now suppose that $\mathcal{R}'$ is not π-fair. Say $\mathcal{R}'$ contains an infinite π-trace $R_k, \ldots, R_{k+1}, \ldots$ starting in N_k.

By property II for π, we find a π-ancestor $P_k \subseteq A_k$ of the π-redex $R_k \subseteq N_k$. (I.e., $\pi(A_k, P_k)$ holds.)

By Proposition 7.6 the π-trace $P_k - \cdot - \cdot \twoheadrightarrow R_k - \cdot - \cdot R_{k+1}$ can be pushed up to go via B_{k+1}; result a π-trace $P_k - \cdot - \cdot \twoheadrightarrow Q_{k+1} - \cdot - \cdot \twoheadrightarrow R_{k+1}$.

Then Q_{k+1} can be traced upward to P_{k+1} in A_{k+1}, while retaining property π and the history repeats itself. After finitely many steps we have found an ancestor P_l of R_l such that $\pi(M_l, P_l)$. Continuing to apply Proposition 7.6, the remainder of the infinite π-trace $R_l - \cdot - \cdot \twoheadrightarrow R_{l+1} - \cdot - \cdot \twoheadrightarrow \cdots$ is transferred to an infinite π-trace $P_l - \cdot - \cdot \twoheadrightarrow P_{l+1} - \cdot - \cdot \twoheadrightarrow$ through $\mathcal{R}$. Hence $\mathcal{R}$ is not π-fair, contradicting our assumption. ∎

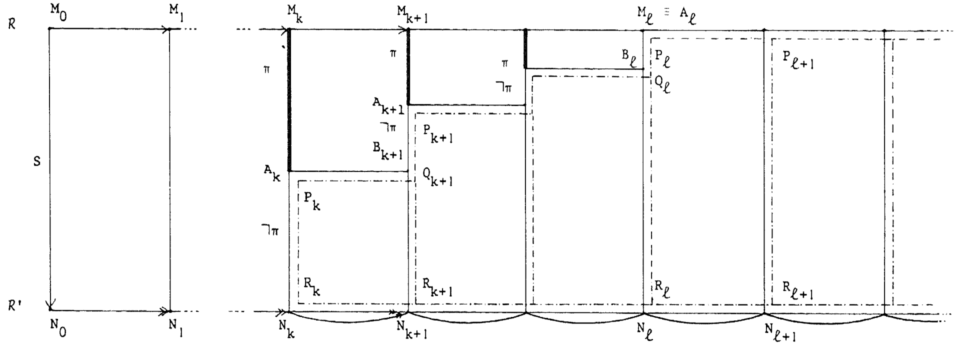

FIGURE 18

7.9. PROPOSITION. *Let $\mathscr{R} = M_0 \to \cdots$ be a reduction containing infinitely many steps in which an outermost redex is contracted. Let $S \subseteq M_0$ be a redex. Then $\mathscr{R}/\{S\}$ is again infinite.*

Proof. The proof for TRSs with bound varables (CRSs) is considerably more complicated than that for ordinary TRSs. Therefore we separate the proofs, even though the one for CRSs entails that for TRSs.

I. *For TRSs* (see Fig. 19). Let $\mathscr{R}$ be as in Proposition 7.9 and suppose $\mathscr{R}' = \mathscr{R}/\{S\}$ is the empty reduction after some M'_k. Consider $l \geqslant k$. If R_l, the redex contracted in $M_l \to M_{l+1}$, is outermost, then the reduction $M'_l \twoheadrightarrow M'_{l+1}$ can only be empty if R_l is one of the residuals of S contracted in $\mathscr{R}_l$. In that case $\mathscr{R}_{l+1}$ has one step less than $\mathscr{R}_l$.

Otherwise, R_l is properly contained in some residual of S contracted in $\mathscr{R}_l$. (Here the proof for the case with bound variables would break down.) Hence since $\mathscr{R}$ contains infinitely many outermost steps, after some q, $\mathscr{R}_q$ is empty. So $\mathscr{R}'$ coincides after M_q with $\mathscr{R}$ and is therefore also infinite. ∎

II. *For CRSs* (see again Fig. 19). The complication is now due to the fact that the residuals S_i of S which are contracted in the development $\mathscr{R}_n$, $n \geqslant 1$, may be nested. Therefore R_n, even when it is a proper subredex in one of the S_i contracted in $\mathscr{R}_n$, may contain some residuals S_j and so may multiply them. Hence $\mathscr{R}_{n+1}$ could have more steps than $\mathscr{R}_n$.

The idea of the following proof is that this does not matter: if R_n is a proper subredex of an S_i, and R_n is not itself a residual of S, then $M'_n \twoheadrightarrow M'_{n+1}$ can only be empty because R_n is *erased* by $\mathscr{R}_n$. That means that $\mathscr{R}_n$ and the S_j contained by R_n are in a "dark spot" of M_n where it does not matter what happens.

We will keep track of the residuals of S in $\mathscr{R}$ by underlining their headsymbol. So each $\mathscr{R}_n$ $(n \geqslant 0)$ is a development of the underlined redexes in M_n.

Let k be as before, in I. In the terms M_l $(l \geqslant k)$ we will distinguish (or rather, obscure) some subterms by surrounding them by a box, as follows. Boxes may be nested, e.g., as in

$$H(\boxed{F(A, G(\boxed{B}))})$$

We will call a subterm in a box "*obscured.*"

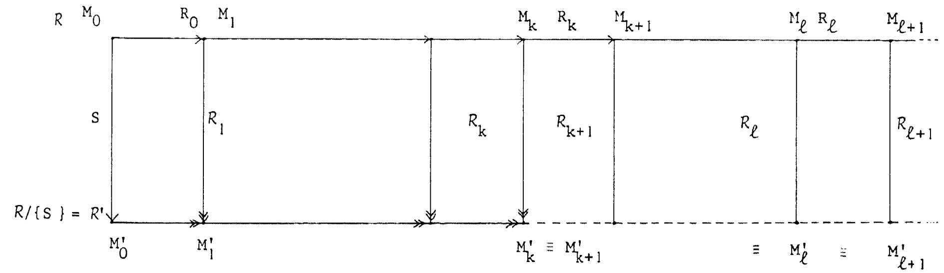

FIGURE 19

Basis Step. In M_{k-1} none of the subterms is obscured.

Induction step. Suppose for M_l we have defined the obscured subterms. Then:

(i) the quasi-descendants (see Definition in 1.4.1.1) in M_{l+1} of those obscured subterms will be again obscured, and

(ii) if R_l is a proper subredex of an underlined redex, and R_l is itself not underlined, then R_l is obscured.

Furthermore, a reduction step in $\mathscr{R}$ is called obscured if it takes place inside a box.

CLAIM 1. *There are only finitely many non-obscured steps in* $\mathscr{R}$.

Proof of Claim 1. Consider the reduction $M_k \to M_{k+1} \to \cdots$ plus boxes and underlining. Replace every outermost box in this reduction by the new symbol $\square$. Result: $\mathscr{R}_\square$. (So now the obscured subterms are really obscure.) Then some of the steps in $\mathscr{R}_\square$ become empty, namely those in which an obscured redex was contracted. In fact only finitely many steps in $\mathscr{R}_\square$ will be non-trivial. This is evident from the finite developments theorem 1.4.3; for, $\mathscr{R}_\square$ is nothing else than a development of underlined redexes in which sometimes subterms are replaced by $\square$. (Note that redexes not covered by an underlined redex cannot be contracted since otherwise the projection of such a contraction would not be empty.) This ends the proof of Claim 1.

CLAIM 2. *Every obscured underlined redex in* $\mathscr{R}$ *is properly contained in a not obscured underlined redex.*

Proof of Claim 2. Suppose not. (See Fig. 20.) Let M_p for some $p \geqslant k$ be a term in $\mathscr{R}$ containing an underlined, obscured redex which is not covered by a non-obscured underlined redex. Choose S_i to be maximal. Note that S_i is a maximal underlined redex.

Now let M_l be the first term in $\mathscr{R}$ where the ancestor of S_i (call it S_i') was obscured. So $S_i' \subsetneq R_l$, and R_l is not underlined. We will devise a development $\mathscr{R}_l'$ of the underlined redexes in M_l such that $R_l' \simeq \mathscr{R}_l$ and S_i' is not contracted in $\mathscr{R}_l'$, as follows.

In $\mathscr{R}_l'$ we contract only (in an arbitrary way) underlined redexes which are not contained by R_l. By the finite developments theorem 1.4.3, this procedure must stop eventually, say in M_l^*. In M_l^* there can be no residual of R_k. For, if there was, this residual would not be covered by an underlined redex; and hence $M_l' \twoheadrightarrow M_{l+1}'$ would not be empty. (In fact, the reduction $M_l' \longrightarrow\!\!\!\!\!\rightarrow M_{l+1}'$ (see Fig. 20), defined as $M_l \to M_{l+1}/\mathscr{R}_l'$ would not be empty; since $\mathscr{R}_l \simeq \mathscr{R}_l'$ we have $M_l' \twoheadrightarrow M_{l+1}' \simeq M_l' \longrightarrow\!\!\!\!\!\rightarrow M_{l+1}'$ and an empty reduction cannot be equivalent to a non-empty one.) Therefore R_k must be *erased* in M_l^*. But then S_i', properly contained by R_l, must also be erased. Hence $\mathscr{R}_l'$ ends in fact in M_l', i.e., $M_l^* \equiv M_l'$. Since all complete developments are equivalent (1.4.3.1), $\mathscr{R}_l' \simeq \mathscr{R}_l$. Now $\mathscr{R}_p = \mathscr{R}_l/M_l \twoheadrightarrow M_p$; and putting $\mathscr{R}_p' = \mathscr{R}_l'/M_l \twoheadrightarrow M_p$ we have, by $\mathscr{R}_l \simeq \mathscr{R}_l'$, the equivalence $\mathscr{R}_p \simeq \mathscr{R}_p'$. Because

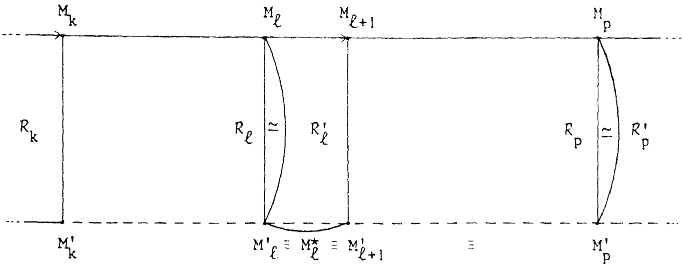

FIGURE 20

$\mathscr{R}'_l$ does not contract S'_i by the parallel moves lemma 1.4.1, $\mathscr{R}'_p$ does not contain steps in which S_i is contracted. But clearly, since S_i was a maximal underlined redex, every complete development of the underlined redexes in M_p must contract S_i. Contradiction. This proves Claim 2. Now let q be such that all steps in $\mathscr{R}$ beyond M_q are obscured (by Claim 1 such a q exists).

CLAIM 3. *In every step $M_{q+j} \to M_{q+j+1}$ $(j \geqslant 0)$ the contracted redex $\mathscr{R}_{q+j}$ is not an outermost redex.*

Proof of Claim 3. Since all steps beyond M_q are obscured, $\mathscr{R}_{q+j}$ is in a box. If R_{q+j} is an underlined redex, it is not outermost by Claim 2.

If R_{q+j} is not underlined and is an outermost redex, a contraction of R_{q+j} results in a non-empty projection $M'_{q+j} \twoheadrightarrow M'_{q+j+1}$, contrary to the assumption for $\mathscr{R}'$. This proves Claim 3.

Claim 3 contradicts the hypothesis of the proposition for $\mathscr{R}$. Hence our assumption that $\mathscr{R}'$ is finite, is false. $\blacksquare_{\text{II}}$

The following corollary is due to O'Donnell [16] for TRSs. (Type I* or III_n^* refers to type I or III_n + bound variables, see Introduction.)

7.10. COROLLARY. *For every type* I* *or* III_n^* *rewriting system*: (i) *Define* $\pi(M, R)$ *by "R is an outermost redex of M." Then the class of π-fair reductions is terminating.*

(ii) *Parallel outermost reductions are terminating.*

Proof. (i) Suppose M_0 has normal form N. (If M_0 has no normal form, there is nothing to prove: by definition, the statement in (i) of the corollary means that the class of π-fair reductions is terminating *whenever possible.*) Let $\mathscr{R} = M_0 \to M_1 \to \cdots$ be an infinite π-fair ("eventually outermost" in [16]) reduction. Obviously $\mathscr{R}$ contains infinitely many outermost steps. Hence $\mathscr{R}'$ (see Fig. 21) is infinite by Proposition 7.9; and π-fair by Theorem 7.8. But continuing in this fashion we find that $\mathscr{R}^{(k)} = \mathscr{R}/M_0 \twoheadrightarrow N$ must be finite, contradicting the fact that N is a normal form.

(ii) Immediately by (i), since evidently a parallel outermost reduction is π-fair. $\blacksquare$

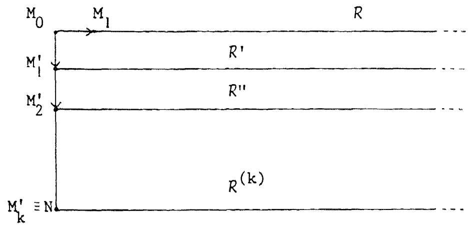

FIGURE 21

7.11. *Leftmost Reductions*

For *leftmost* reductions, in which each time the leftmost redex (that is, the redex whose head symbol is leftmost) is contracted, the analogous corollary fails.

EXAMPLE. (from [10]). Let Σ be a TRS having as rule schemes:

$$F(x, B) \to D, \qquad A \to B, \qquad C \to C.$$

Then $F(C, A) \to F(C, A) \to \cdots$ (each step a contraction of redex C) is a counterexample.

However, if Σ is a "left-normal" system, one can prove that (eventually) leftmost reductions are normalizing. This was done in [14] via a standardization method; the proof we will give below is more perspicuous and is, for TRSs, given in [16]. We will again derive the result for TRSs where bound variables may be present, in fact for type I* or III_n^* systems.

7.12. DEFINITION. (i) Let Σ be a regular CRS, and let r be a rule in Σ; $r = H \to H'$. Then r is left-normal if in H all operator symbols (including the 0-ary operators, i.e., the constants) precede the variables. E.g., the rule $F(x, B) \to D$ above is not left-normal; the rule $F(B, x) \to D$ is left-normal.

(ii) Σ is left-normal iff all its rules are left-normal.

(iii) If Σ is a type I* or III_n^* system, Σ is left-normal iff Σ_u is.

7.13. COROLLARY. *Let Σ be of type* I* *or* III_n^* *and left-normal. Then for Σ-reductions*:

(i) *eventually leftmost reductions are terminating*

(ii) *the leftmost reduction is terminating.*

Proof. Let $\pi(M, R)$ be: R is the leftmost redex in M. Then property I and II (Definition 7.2) are easily verified for π (for II we need the left-normality). Hence by Theorem 7.8, π-fair reductions (i.e., eventually leftmost reductions) are closed under projections. Furthermore, Proposition 7.9 is valid for "leftmost" instead of "outermost" because the leftmost redex is outermost. Hence the result follows. ∎

7.14. EXAMPLE. (i) For λ-calculus + "recursor" R having the rule schemes $Rxy0 \to x$, $\mathscr{R}xy(Sz) \to xz(Rxyz)$ we have termination of parallel outermost reductions—but not of the leftmost reduction strategy.

(ii) For λ-calculus + alternative recursor R', such that $R'0xy \to x$, $R'(Sz)xy \to xz(R'xyz)$ also the leftmost reduction strategy is terminating.

(iii) For the system in (i) one can obtain a slightly better result than termination of parallel outer most reductions, by introducing O'Donnell's "*dominance ordering*," an extension of the subterm ordering ($\subseteq$), which would in this case cause the redexes in the third argument of R to be priviliged above those in the first two arguments.

7.15. EXAMPLE. If Σ is the type III_n reduction system corresponding to BOUNDED-STACK (see Example 2.1(i)) then Σ is left-normal. Hence the results above yield that both parallel outermost reduction and leftmost reduction terminate whenever possible. (In this case that is trivial since all reductions terminate, as one easily proves.)

ACKNOWLEDGMENT

We thank Klaus Drosten for detecting an error in a previous version of Definition 4.2 and suggesting a correction. Also, we thank a referee for many suggestions for improvements, including a rearrangement of the paper in an earlier version.

REFERENCES

1. H. P. BARENDREGT, "The Lambda Calculus, Its Syntax and Semantics," 2nd rev. ed., North-Holland, Amsterdam, 1984.
2. J. A. BERGSTRA AND J. V. TUCKER, A characterisation of computable data types by means of a finite, equational specification method, *in* "Proc. of 7th ICALP, Noordwijkerhout 1980" (J. W. de Bakker and J. van Leeuwen, Eds.), Lecture Notes in Comput. Sci. Vol. 85, pp. 76–90, Springer, New York, 1980.
3. G. BERRY AND J. J. LÉVY, Minimal and optimal computations of recursive programs, *J. Assoc. Comput. Mach.* **26** (1979), 1.
4. P. CHEW, Unique normal forms in term rewriting systems with repeated variables, *in* "Proc. of 13th Annual ACM Symposium on the Theory of Computing," pp. 7–18, 1981.
5. H. B. CURRY AND R. FEYS, "Combinatory Logic," Vol. I, North-Holland, Amsterdam, 1958.
6. K. DROSTEN, "Toward Executable Specifications Using Conditional Axioms," Report 83-01, T. U. Braunschweig, 1983.
7. H. E. EHRIG, H.-J. KREOWSKI, J. W. THATCHER, E. G. WAGNER, AND J. B. WRIGHT, Parameterized data types in algebraic specification languages, *in* "Proc. 7th ICALP, Noordwijkerhout 1980" (J. W. de Bakker and J. van Leeuwen, Eds.), Lecture Notes in Comput. Sci. Vol. 85, Springer, New York, 1980.
8. J. A. GOGUEN, J. W. THATCHER, AND E. G. WAGNER, An initial algebra approach to the specification, correctness and implementation of abstract data types, *in* "Current Trends in Programming Methodology. IV, Data Structuring," (R. T. Yeh, Ed.), Prentice-Hall, Engelwood Cliffs, N.J., 1978.

9. G. HUET, Confluent reductions: abstract properties and applications to term rewriting systems, *in* "18th IEEE Symposium on Foundations of Computer Science, 1977," pp.. 30–45; rev. version: IRIA Report, Jan. 1978.
10. G. HUET AND J. J. LÉVY, "Call by Need Computations in Non-ambiguous Linear Term Rewriting Systems," Rapport Laboria 359, IRIA, 1979.
11. G. HUET AND D. C. OPPEN, Equations and rewrite rules: A survey, *in* "Formal Languages: Perspective and Open Problems" (R. Book, Ed.), Academic Press, New York/London, 1980.
12. S. KAPLAN, Conditional rewrite rules, *Theor. Comput. Sci.*, in press.
13. S. KAPLAN, "Fair Conditional Term Rewriting Systems: Unification, Termination and Confluence, Rapport de Récherche No. 194, Univ. de Paris-Sud, LRI Orsay, 1984.
14. J. W. KLOP, "Combinatory Reduction Systems," Mathematical Centre Tracts Vol. 127, Centre for Mathematics and Computer Science, Amsterdam, 1980.
15. J. J. LÉVY, "Réductions correctes et optimales dans le lambda-calcul," Thèse d'Etat, Univ. de Paris VII, 1978.
16. M. O'DONNELL, "Computing in Systems Described by Equations," Lecture Notes in Comput. Sci. Vol. 58, Springer, New York, 1977.
17. M. O'DONNELL, "Equational Logic as a Programming Language," MIT Press, Cambridge, Mass., 1985.
18. U. PLETAT, G. ENGELS, AND H.-D. EHRICH, Operational semantics of algebraic specifications with conditional equations, *in* "Proc. 7th CAAP," Lille 1981, Lecture Notes in Comput. Sci. Vol. 112, Springer, New York, 1981; extended version: Forschungsbericht 118/81, Univ. Dortmund, 1981.