III. SUBTREE REPLACEMENT SYSTEMS

1. A Formalism for Reductions

We are interested in sets A of equational axioms which may be treated as reduction rules because each expression on the right-hand side is simpler or clearer than that on the left. Thinking of A as a set of reduction rules, a reduction relation, $\rightarrow$, may be defined as follows:

$$A \rightarrow B \text{ iff } \exists_{x \in \underline{\text{domain}}A,\, C \in \Sigma_*} \langle A/x=C \rangle \in A \wedge B = A(x \leftarrow C)$$

$A \rightarrow B$ means that, for some x and C, A=B follows from the axiom A/x=C by one application of the substitution rule for equality.

Definition 16

A Subtree Replacement System (SRS) is a 4-tuple $\langle \Sigma, F, \rightarrow, A \rangle$ where:

- Σ is any set
- $F \subseteq \Sigma_*$ is a set (forest) of Σ-trees
- $A \subseteq \Sigma_* \times \Sigma_*$ is a set of ordered pairs of Σ-trees (think of the equation A=B as a special notation for the pair $\langle A,B \rangle$)
- $\rightarrow$ is a binary relation on Σ-trees

For all Σ-trees A and B, $A \rightarrow B$ iff

$$\exists_{x \in \underline{\text{domain}}A,\, C \in \Sigma_*} \langle A/x=C \rangle \in A \wedge B = A(x \leftarrow C)$$

F is closed under $\rightarrow$, i.e.,

$$(A \in F \wedge A \rightarrow B) \Rightarrow B \in F$$
$$(A \in F \wedge B \rightarrow A) \Rightarrow B \in F$$

A is a partial function on Σ_*, i.e.,

$$(\langle A=B \rangle \in A \wedge \langle A,C \rangle \in A) \Rightarrow B = C.$$

The definition above is the same as Rosen's definition of an unequivocal Subtree Replacement System [Ro73] (Defs.5.1, 5.2, pp. 169, 170). Intuitively, a SRS consists of a set of expressions F using symbols from Σ (F will often be $\Sigma_\#$), with equations A defining a reduction relation $\rightarrow$. Because of the last clause, only sets A of equations in which left-hand sides are unique may be associated with SRSs.

Example 6

Let $\Sigma = \{0, s, p, Cond, F\}$ as in Example 2.

To represent the recursive definition:

$$F(x,y) = Cond(x,0,F(p(x)),F(x,y))$$

with the axioms for s, p, Cond:

$$p(s(x)) = x$$
$$Cond(0,x,y) = x$$

$\mathrm{Cond}(s(x),y,z) = z,$

we let $A = \{<F(A,B) = \mathrm{Cond}(A,0,F(p(A),F(A,B)))>,$
$<p(s(A)) = A>,$
$<\mathrm{Cond}(0,A,B) = A>,$
$<\mathrm{Cond}(s(A),B,C) = C> \mid A,B,C \in \Sigma_\#\},$

and let $\rightarrow$ be defined as in Definition 16 (p. 18).

Then $E = <\Sigma,\Sigma_\#,\rightarrow,A>$ is a SRS representing the recursive definition and axioms above.

By Lemma 2 (p.12), if $<\Sigma,F,\rightarrow,A>$ is a SRS and $\equiv$ is the reflexive, symmetric transitive closure of $\rightarrow$, then $A \models A{=}B$ iff $A \vdash A{=}B$ iff $A{\equiv}B$.

2. Sequences of Reductions - Informal Treatment

Given a SRS $<\Sigma,F,\rightarrow,A>$, and an expression $A_0 \in F$, it is reasonable to generate information about A_0 by finding a sequence of expressions A_i such that $A_0 \rightarrow A_1 \rightarrow \ldots \rightarrow A_n$. Each such A_i will have the property that $A \models A_0{=}A_i$, and A_i is a simpler or clearer expression than A_0. If A_n cannot be reduced further, then it is a normal form for A_0.

Example 7

In the SRS E of Example 6 (p.18),

$\mathrm{Cond}(s(s(0)),\ s(0),\ p(p(s(0)))) \rightarrow p(p(s(0))) \rightarrow p(0)$

and $p(0)$ is in normal form.

Figure 4 shows the reduction sequence above.

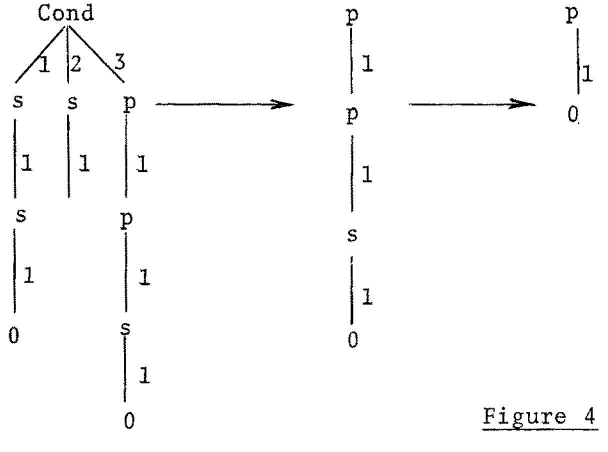


Figure 4

If an expression has several reduction sequence, we would like to know how they are related: whether they yield different normal forms, and which sequence is easiest to compute. To approach these questions we will need a precise notation for describing different reduction steps and reduction sequences.

3. A Formal Notation for Reduction Steps and Sequences

Since $\mathcal{A}$ represents a partial function, a reduction step $A_i \rightarrow A_{i+1}$ is completely specified by naming the address of the subtree in A_i which is replaced.

Definition 17

x is a *redex* in A iff $\exists_B \langle A/x{=}B\rangle \in \mathcal{A}$. That is, x is a redex in A whenever A may be reduced by replacing the subtree at x.

Example 8

In the SRS $\mathcal{E}$ of Example 6 (p. 18), the expression s(Cond(s(0), p(s(0)), F(0,0))), shown in Figure 5,

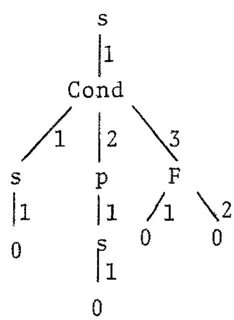


Figure 5

has redexes (1), (1,2), (1,3).

Definition 18

$A \xrightarrow[x]{} B$ iff x is a redex in A and $B = A(x \leftarrow C)$ where $\langle A/x{=}C\rangle \in \mathcal{A}$. $A \xrightarrow[x]{} B$ means that A may be reduced to B by replacing the subtree at x in A.

Example 9

In $\mathcal{E}$ of Example 6 (p. 18),
s(Cond(s(0), p(s(0)), F(0,0))) $\xrightarrow[(1,2)]{}$ s(Cond(s(0), 0, F(0,0))).
Figure 6 shows the reduction above.

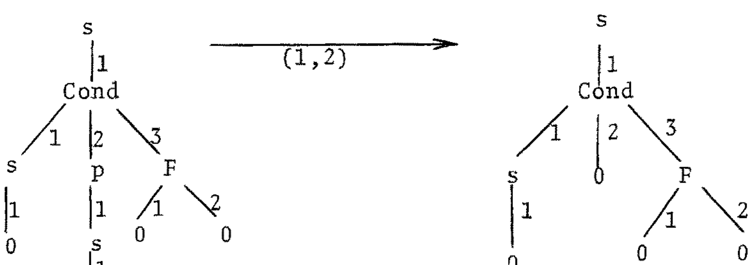


Figure 6

Notice that; $\xrightarrow[x]{} \subseteq \rightarrow$; $\xrightarrow[()]{} = A$; $\xrightarrow[x]{}$ is a partial function.

It is convenient and reasonable in some contexts to treat a set of independent reductions as a single step.

<u>Definition 19</u>

Let $M \in P^{\perp}$(<u>domain</u>A) be an independent set of redexes in A, then

$$A \xrightarrow[M]{} B \text{ iff } B = A(x \leftarrow C_x \mid x \in M) \text{ where } \forall_{x \in M} \langle A/x = C_x \rangle \in A$$

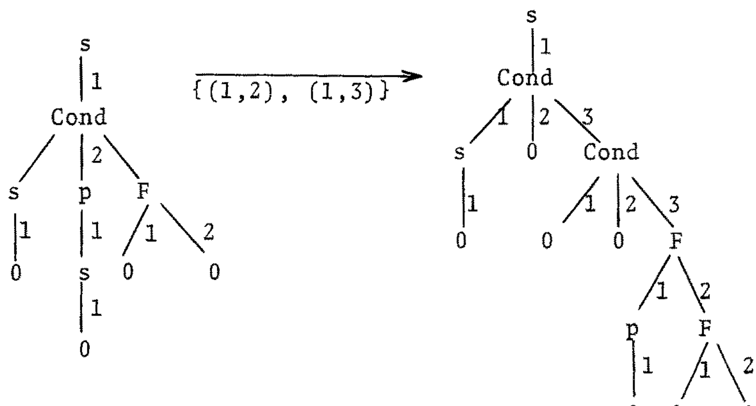
$A \xrightarrow[M]{} B$ means that A may be reduced to B by replacing each subtree at $x \in M$ in A. Since M is an independent set, the order of replacements is irrelevant.

<u>Example 10</u>

In E of Example 6 (p. 18),

$$s(Cond(s(0), p(s(0)), F(0,0))) \xrightarrow[\{(1,2),\ (1,3)\}]{} s(Cond(s(0),0,Cond(0,0,F(p(0),F(0,0))))).$$

Figure 7 shows the reduction above.



<u>Figure 7</u>

Notice that $A \xrightarrow[\phi]{} A$; $\xrightarrow[M]{} \subseteq \rightarrow^{|M|}$; $\xrightarrow[\{x\}]{} = \xrightarrow[x]{}$; $\xrightarrow[M]{}$ is a partial function. If M and N are mutually independent, then $\xrightarrow[M]{}$, $\xrightarrow[N]{}$ commute.

<u>Lemma 5</u>

If $M \perp N$ and $A \xrightarrow[M]{} B$, $A \xrightarrow[N]{} C$, then there is a D such that $B \xrightarrow[N]{} D$, $C \xrightarrow[M]{} D$, $A \xrightarrow[M \cup N]{} D$. Figure 8 summarizes the Lemma.

$$\begin{tikzcd}
& A \arrow[dl, "M"'] \arrow[dr, "N"] \arrow[dd, "M \cup N"] & \\
B \arrow[dr, "N"'] & & C \arrow[dl, "M"] \\
& D &
\end{tikzcd}$$

<u>Figure 8</u>

Proof   Simple induction on $|M|+|N|$, using Definition 19. □

Sometimes we will not need to specify a reduction step $A \xrightarrow[N]{} B$ completely, but must know that it is the result of independent replacements.

Definition 20

$A \xrightarrow[\perp]{} B$ iff there is an $M \in P^{\perp}$(domainA) such that $A \xrightarrow[M]{} B$.

Since the order in which reductions are performed on an independent set of redexes is irrelevant, we will write reduction sequences in a way that ignores this order.

Definition 21

The sequence $(A_0, M_1, A_1, M_2, A_2, \ldots)$ is a reduction sequence iff for all A_i and M_i in the sequence, $A_i \xrightarrow[M_i]{} A_{i+1}$. If A_n is in normal form, then the reduction sequence $(A_0, M_1, A_1, \ldots, M_n, A_n)$ is a reduction of A_0 to normal form A_n.

Notice that a reduction sequence may be infinite. Reduction sequences may be written $A_0 \xrightarrow[M_1]{} A_1 \xrightarrow[M_2]{} \ldots$ or simply (A_i) when convenient. The value of reduction sequences is that they closely mimic a large class of computations which seek normal forms for expressions, and also correspond directly to equational proofs.

4. Residuals

Interesting properties of reduction sequences may be affected by the way in which redexes are created, preserved and destroyed by reduction steps. In many SRSs which arise naturally, the replacement of A by B where $\langle A=B \rangle \in A$ may be viewed as a rearrangement of A. For instance, the replacement of a*(b+c) by (a*b)+(a*c) according to the distributive axiom is really a rearrangement of the parts of the expression a*(b+c). When an expression is rearranged, some subexpressions may be preserved or repeated, as the b and c above are preserved and the a is repeated. When these subexpressions are redexes, their preservation or repetition may be crucial to the properties of a reduction sequence. A residual map r is intended to illuminate the rearrangement process by mapping addresses x in an expression A to sets of addresses $r\langle A=B \rangle x$ in B which are copies of x under the rearrangement. The following definition is a generalization of Rosen's definition of a residual map [Ro73] (Def. 5.3, p.170).

Definition 22

A residual map is a function $r:A\to(P^*\to P^{\perp}(P^*))$ such that

(1) (x is a redex in A $\wedge$ $y\in r\langle A=B\rangle x$) $\Rightarrow$ y is a redex in B

(2) $x\perp y \Rightarrow r\langle A=B\rangle x \perp r\langle A=B\rangle y$

(3) $r\langle A=B\rangle () = \phi$

Clause (1) says that residuals of redexes are redexes, clause (2) that residuals of independent redexes are independent, and clause (3) says that the redex which is reduced disappears.

Example 11

In E of Example 6 (p. 18), we may define a residual map r as follows:

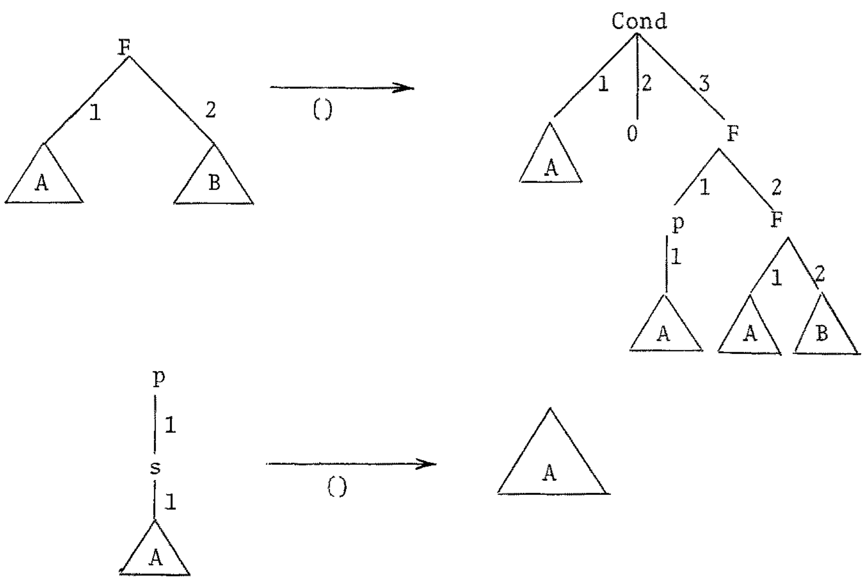
$$r\langle F(A,B)=\mathrm{Cond}(A,0,F(p(A),F(A,B)))\rangle x = \begin{cases} \{(1)\cdot z,\ (3,1,1)\cdot z,\ (3,2,1)\cdot z\} & \text{if } x=(1)\cdot z \\ \{(3,2,2)\cdot z\} & \text{if } x=(2)\cdot z \\ \phi & \text{otherwise} \end{cases}$$

$$r\langle p(s(A))=A\rangle x = \begin{cases} \{z\} & \text{if } x=(1,1)\cdot z \\ \phi & \text{otherwise} \end{cases}$$

$$r\langle \mathrm{Cond}(0,A,B)=A\rangle x = \begin{cases} \{z\} & \text{if } x=(2)\cdot z \\ \phi & \text{otherwise} \end{cases}$$

$$r\langle \mathrm{Cond}(s(A),B,C)=C\rangle x = \begin{cases} \{z\} & \text{if } x=(3)\cdot z \\ \phi & \text{otherwise} \end{cases}$$

Looking at Figure 9,



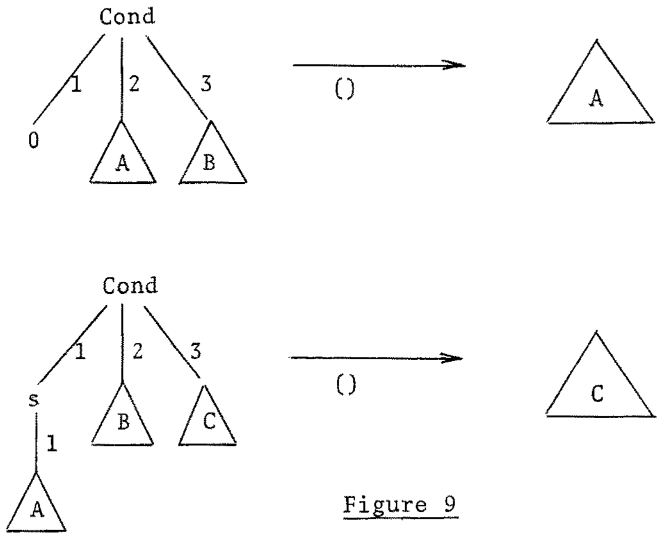


Figure 9

we see that in all cases, the residuals of an address x in a subtree A, B, or C on the left-hand side are the corresponding addresses in the copies of subtrees A, B, C on the right. Other addresses have no residuals. For instance, in Figure 10 (1,1) and its residuals have been marked with a *.

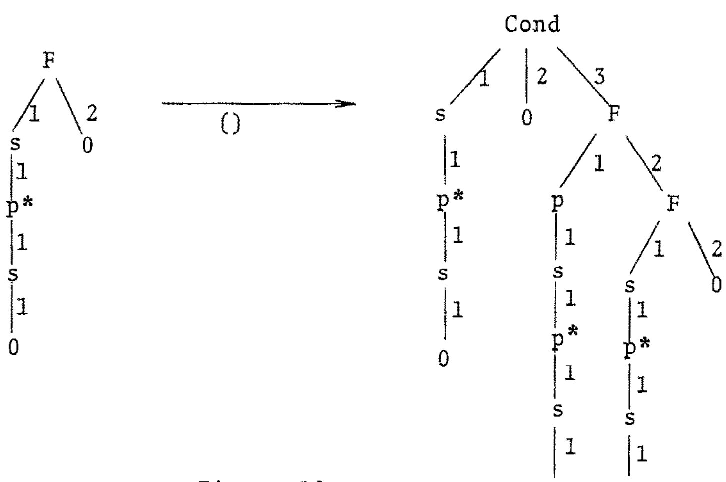


Figure 10

If A is the left-hand expression, and B the right, then r<A=B>(1,1) = {(1,1), (3,1,1,1), (3,2,1,1)}.

The residual map r shows how redexes are rearranged by a single reduction step. The <u>extended residual map</u> $\hat{r}$ shows how redexes are rearranged by a sequence of reductions.

Definition 23

If r is a residual map then the extended residual map $\hat{r}$ is a function $\hat{r}: F\times(P^{\perp}(P^{*}))^{*} \to (P^{\perp}(P^{*}) \to P^{\perp}(P^{*}))$ such that

$$\hat{r}[A,()]\{y\} = \hat{r}[A,(\phi)]\{y\} = \{y\}$$

$$\hat{r}[A,(\{x\})]\{y\} = \begin{cases} \{x\cdot w \mid w\in r<A/x=C>z\} & \text{if } <A/x=C>\in A \text{ and } y=x\cdot z \\ \{y\} & \text{otherwise} \end{cases}$$

$$\hat{r}[A,(\{x\})]N = \bigcup_{y\in N} [\hat{r}[A,(\{x\})]\{y\}$$

$$\hat{r}[A,(M_1,\ldots,M_{n+1})]N = r[B,(M_{n+1})](r[A,(M_1,\ldots,M_n)]N)$$

$$\text{where } A \xrightarrow[M_1]{} \cdots \xrightarrow[M_n]{} B$$

$$\hat{r}[A,(M)]N = \hat{r}[A,(\{x_1\},\ldots,\{x_m\})]N$$

$$\text{where } M = \{x_1,\ldots,x_m\}$$

$\hat{r}$ is well-defined by a straightforward argument which hinges on the fact that the M, M_i and N are independent sets, and that $A, \xrightarrow[M_i]{}$ are partial functions. Intuitively, $\hat{r}[A,(M_1,\ldots,M_n)]N$ gives the residuals of all tree addresses $x\in N$ as they are rearranged by the reductions $A \xrightarrow[M_1]{} \cdots \xrightarrow[M_n]{} B$. The extended residual map $\hat{r}$ allows us to keep track of all the rearranged copies of a redex through an arbitrarily long reduction sequence.

Example 12

Consider E of Example 6 (p.18), r defined in Example 11 (p.23). In Figure 11 on the next page, (1,1,1) and its extended residuals have been marked with a *.

Notice that when the addresses N in A are rearranged by reductions at the redexes M, all residuals in $\hat{r}[A,(M)]N$ are descendants of addresses in M and N.

Lemma 6

Let $\hat{r}$ be an extended residual map, then

$$\forall_{x\in\hat{r}[A,(M)]N}\ \exists_{y\in M\cup N}(y \underline{\text{ anc }} x \wedge \exists_{z\in N}\ y \underline{\text{ anc }} z)$$

Proof Consider $z\in N$. If z has no ancestor in M, then $\hat{r}[A,(M)]\{z\} = \{z\}$. If z has an ancestor $w\in M$, then all of $\hat{r}[A,(M)]\{z\}$ have w as an ancestor also. □

Lemma 6 and the following two Lemmas on residuals will be useful in Chapters IV and V.

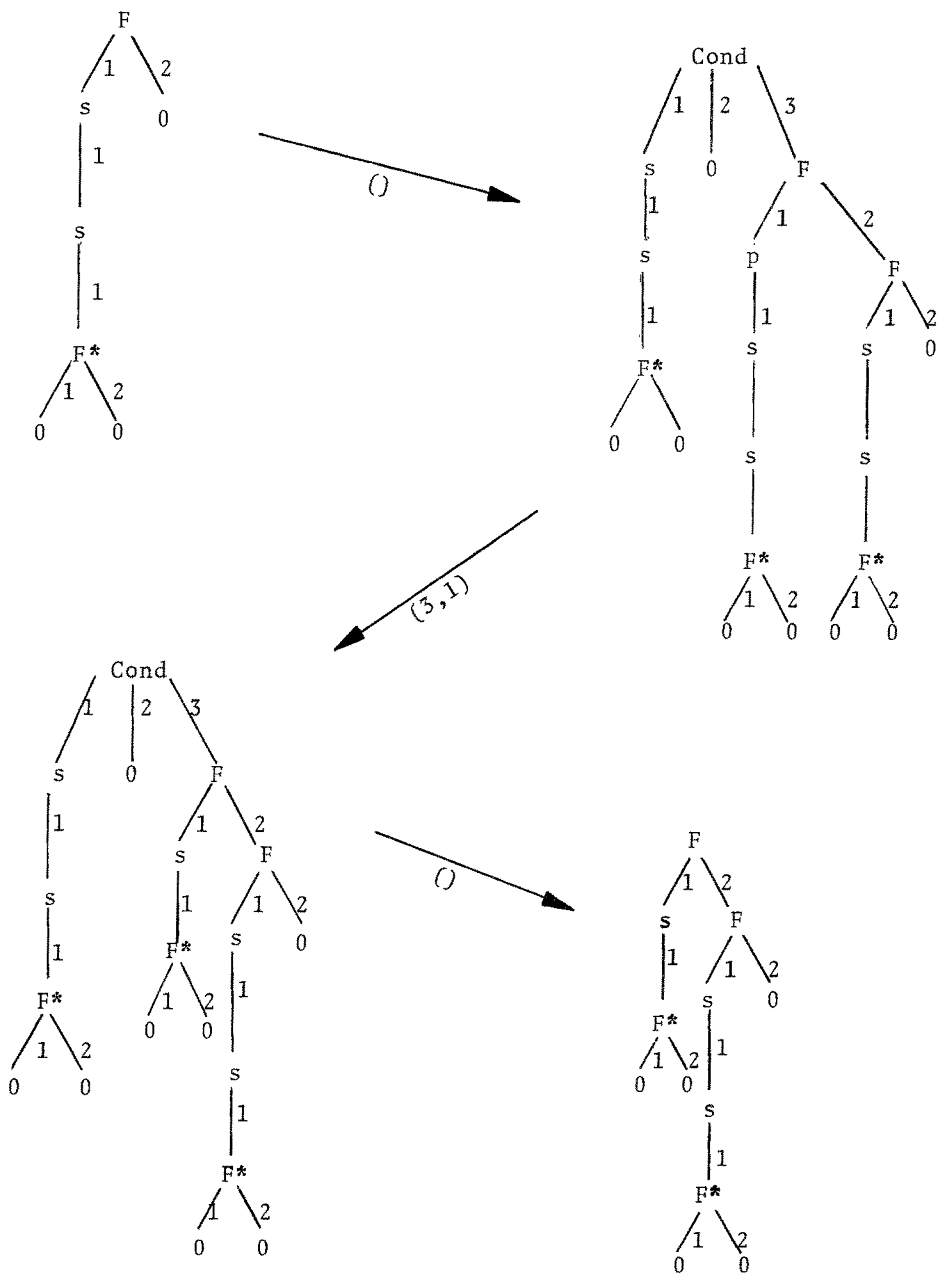

F
1
2
s
0
1
s
1
F*
1
2
0
0
()
Cond
1
2
3
s
0
F
1
1
2
s
p
F
1
1
1
2
F*
s
s
0
0
0
s
s
F*
F*
1
2
1
2
0
0
0
0
(3,1)
Cond
1
2
3
s
0
F
1
1
2
s
s
F
1
1
1
2
F*
F*
s
0
1
2
1
1
2
0
0
0
0
s
1
F*
1
2
0
0
()
F
1
2
s
F
1
1
2
F*
s
0
1
2
1
0
0
s
1
F*
1
2
0
0

Figure 11

Lemma 7

Let $\hat{r}$ be an extended residual map, and $M_1 \cup N_1 \perp N_2$, $M_2 \cup N_2 \perp N_1$. Then $\hat{r}[A,(M_1)]N_1 \perp \hat{r}[A,(M_2)]N_2$.

Proof Direct from Lemma 6. $\square$

Lemma 8

Let r be an extended residual map.

Then (1) $\hat{r}[A,(M_1,\ldots,M_n)]N_1 \cup \hat{r}[A,(M_1,\ldots,M_n)]N_2$
$= \hat{r}[A,(M_1,\ldots,M_n)](N_1 \cup N_2)$

(2) $N_1 \perp N_2 \Rightarrow \hat{r}[A,(M_1,\ldots,M_n)]N_1 \perp \hat{r}[A,(M_1,\ldots,M_n)]N_2$

(3) $N \not\subseteq M \Rightarrow \hat{r}[A,M]N = \phi$.

Proof An easy induction on $|M_1|+\ldots+|M_n|$, $|N_1|+|N_2|$. $\square$

Clauses (2) and (3) generalize clauses (2) and (3) of Definition 22 (p. 23).