VI. COSTS OF REDUCTION SEQUENCES

1. A Cost Measure for Noncopying Sequences

If an expression has several reduction sequences which end in normal form, we would like to compare their computational costs. An easy measure of the cost of a reduction sequence $A_0 \xrightarrow[M_1]{} A_1 \xrightarrow[M_2]{} \cdots \xrightarrow[M_n]{} A_n$ is the number of nonempty redex sets M_i in the sequence. This measure is reasonable if all nonempty reduction steps $A_i \xrightarrow[M_{i+1}]{} A_{i+1}$ in all allowable sequences have roughly the same cost when implemented.

The cost of a reduction step depends critically on details of the representation of expressions and the implementation of reduction steps. In many interesting SRSs it is the case that, for all y∈r<A=B>x, B/y=A/x. In such cases, a clever programmer is likely to represent expressions by acyclic pointer structures. Reduction steps will not cause the subexpression A/x to be copied many times, but each occurrence of A/x at an address y∈r<A=B>x in B will be represented by a pointer to a single structure.

Example 17

In $\mathcal{E}$ of Example 6 (p. 18), with r of Example 11 (p. 23), it is always the case that y∈r<A=B>x ⇒ B/y=A/x. For instance,
r<F(0,s(0)) = Cond(A,0,F(p(0),F(0,s(0))))>(1) = {(1),(3,1,1),(3,2,1)}
Cond(A,0,F(p(0),F(0,s(0))))/(3,1,1) = F(0,s(0))/(1) = 0.
The reduction of Figure 24 could be represented by the transformation of pointer structures in Figure 25. See also Appendix A.

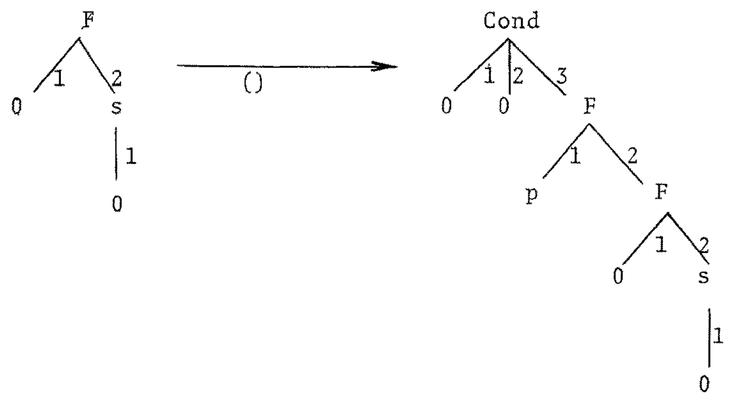


Figure 24

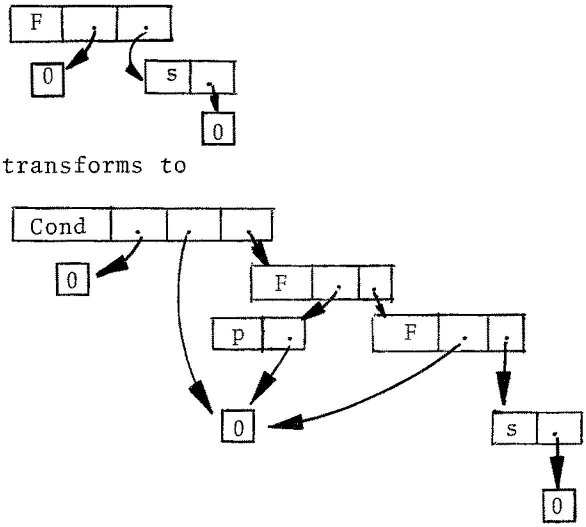


Figure 25

Implementations of reduction without copying of residuals, such as the one hinted at in Example 17, are constrained to reduce all members of a residual set simultaneously. The cost of reducing an entire residual set is independent of the size of the set, since all members of the set are represented by a single structure. The reduction sequences which may be produced by pointer implementations without copying of residuals are called <u>noncopying sequences</u>. Assuming that the method for choosing a redex to reduce at each step is very simple, and that all reductions have approximately the same difficulty, the length of a noncopying sequence is a reasonable measure of its cost. To facilitate future inductions, we allow empty steps to appear in a noncopying sequence at no cost.

<u>Definition 37</u>

If r is a residual map, then the associated <u>equivalent address map</u> is a function $e\colon \mathcal{R} \rightarrow (P^* \rightarrow \mathcal{P}^{\perp}(P^*))$ such that

$$e\langle A=B\rangle x = \begin{cases} x \text{ if } x=() \\ r\langle A=B\rangle x \quad \text{otherwise} \end{cases}$$

Extend e to the <u>extended equivalent address map</u> $\hat{e}$ just as r was extended to $\hat{r}$ (Def. 23, p. 25).

$\hat{e}:F\times(P^{\perp}(P^{*}))^{*}\rightarrow(P^{\perp}(P^{*})\rightarrow P^{\perp}(P^{*}))$ such that

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

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

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

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

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

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

where $M = \{x_1,\dots,x_m\}$

The reduction sequence $A_0 \xrightarrow[M_1]{} A_1 \xrightarrow[M_2]{} \cdots$ is a noncopying sequence iff the following hold for all nonempty M_j:

(1) There is an $i\leq j$ and a redex x in A_i such that $M_j = \hat{e}[A_i,(M_{i+1},\dots,M_{j-1})]\{x\}$.

(2) M_j cannot be expanded without violating Clause (1).

The cost of the noncopying sequence $A_0 \xrightarrow[M_1]{} A_1 \xrightarrow[M_2]{} \cdots$ (written cost(A_i)) is the number of nonempty redex sets M_i in the sequence.

We will restrict attention to noncopying sequences.

Example 18

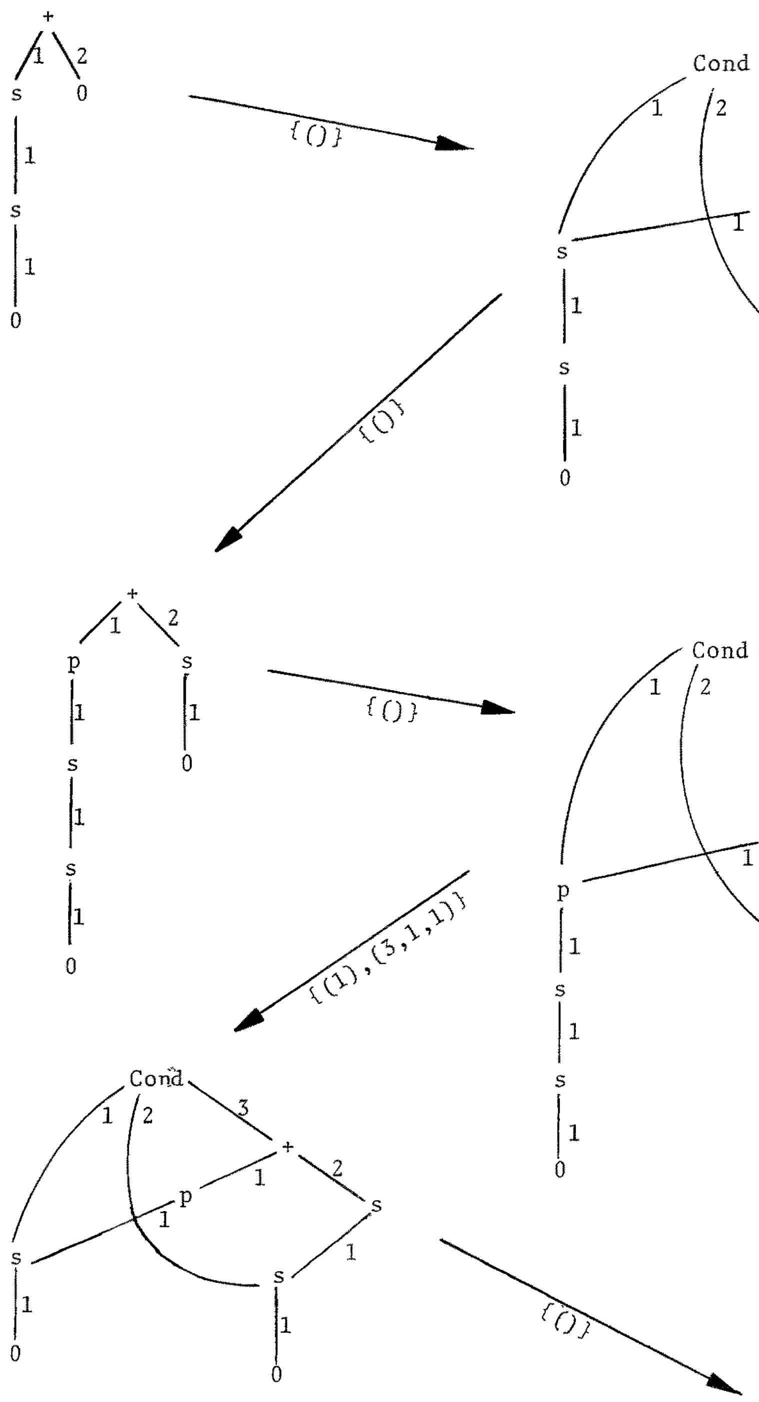
Consider E of Example 6 (p. 18). Add the binary symbol + to Σ, and the equations

{<+(A,B) = Cond(A, B, +(p(A),s(B)))>| A,B∈Σ} to A.

+(s(s(0)),0) may be reduced to s(s(0)) by the following noncopying sequence:

$$
\begin{aligned}
&+(s(s(0)),0) \xrightarrow[\{()\}]{} \text{Cond}(s(s(0)),\ 0,\ +(p(s(s(0))),s(0)))\\
&\xrightarrow[\{()\}]{} +(p(s(s(0))),s(0))\\
&\xrightarrow[\{()\}]{} \text{Cond}(p(s(s(0))),\ s(0),\ +(p(p(s(s(0)))),\ s(s(0))))\\
&\xrightarrow[\{(1),(3,1,1)\}]{} \text{Cond}(s(0),\ s(0),\ +(p(s(0)),s(s(0))))\\
&\xrightarrow[\{()\}]{} +(p(s(0)),s(s(0)))\\
&\xrightarrow[\{()\}]{} \text{Cond}(p(s(0)),\ s(s(0)),\ +(p(p(s(0))),\ s(s(s(0)))))\\
&\xrightarrow[\{(1),(3,1,1)\}]{} \text{Cond}(0,\ s(s(0)),\ +(p(0),\ s(s(s(0)))))\\
&\xrightarrow[\{()\}]{} s(s(0))
\end{aligned}
$$

The sequence above, with equivalent addresses collapsed, is shown in Figure 26.



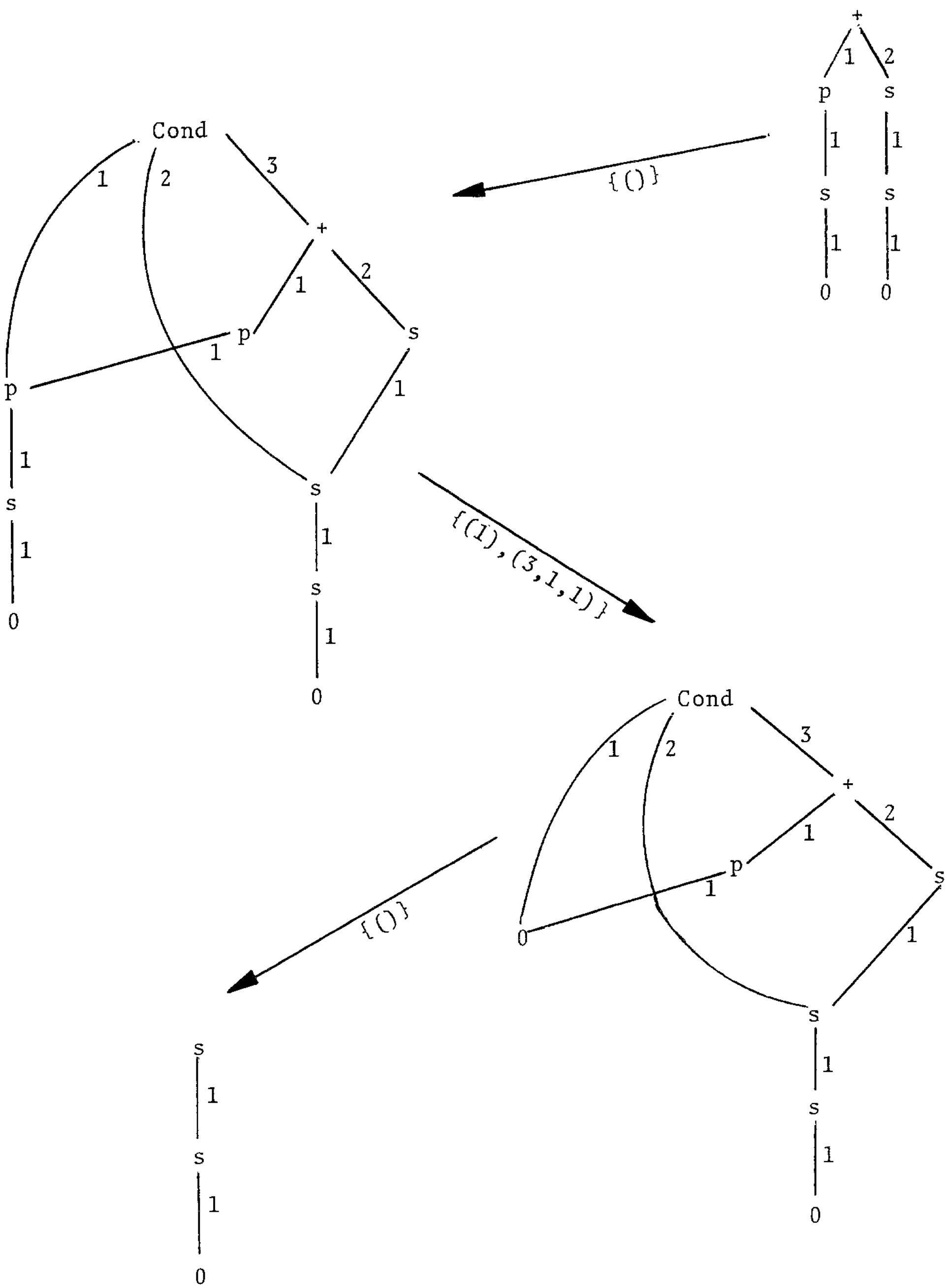


Figure 26

Lemma 19

Let A_0 have normal form D. Then there is a noncopying sequence (A_i) ending in D.

Proof Straightforward. □

2. Commutativity of Residuals

In order to perform interesting manipulations of noncopying sequences we must know that different parts of the sequence may be changed independently. This requires an additional property of the SRS to guarantee that equivalent addresses do not depend on the order of reductions in a sequence.

Definition 38

Let $\mathcal{E} = \langle \Sigma, \mathcal{F}, \to, \mathcal{A} \rangle$ be closed with respect to r.
$\mathcal{E}$ commutes with respect to r iff:
for all $\langle A=B \rangle \in \mathcal{A}$, x and y redexes in A,
$\hat{r}[A,(\{x\},\{()\})]\{y\} = \hat{r}[A,(\{()\},r\langle A=B \rangle x)]\{y\}$

Commutativity is a strengthened form of clause (2) of the definition of closure (Def. 25, p. 30), and is easily verified in many interesting cases (see Chapter VII). The definition of commutativity, applied inductively, shows that any pair of reduction steps may be reordered without affecting equivalent addresses.

Lemma 20

Let $\langle \Sigma, \mathcal{F}, \to, \mathcal{A} \rangle$ commute with respect to r, and let $A \underset{M}{\to} B$ and $A \underset{N}{\to} C$. Then for all sets P of redexes in A,

$$\hat{e}[A,(M,\hat{r}[A,(M)]N)]P = \hat{e}[A,(N,\hat{r}[A,(N)]M)]P.$$

Intuitively, either path in Figure 27 yields the same equivalent addresses.

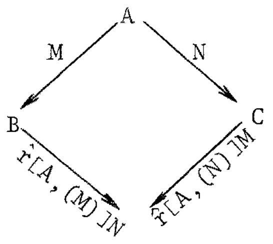


Figure 27

Proof By induction on $|M|$, $|N|$ and $|P|$, using Definition 38. □

Noncopying sequences are closed under innermost reduction, and the cost of a sequence is lessened by at most one due to innermost reduction.

Lemma 21

Let $<\Sigma,F,\rightarrow,A>$ commute with respect to an innermost preserving residual map r, and let $A_0 \xrightarrow[M_1]{} A_1 \xrightarrow[M_2]{} \ldots$ be a noncopying sequence, and let $(A_i) \xrightarrow[(N_i)]{} (B_i)$ where $N_0 \subseteq \underline{imr}A_0$ and $|N_0| \leq 1$. Then $B_0 \xrightarrow[\hat{r}[A_0,(N_0)]M_1]{} B_1 \xrightarrow[\hat{r}[A_1,(N_1)]M_2]{} \ldots$ is a noncopying sequence, and $\underline{cost}(B_i) \geq \underline{cost}(A_i)-1$.

Proof M_1 has at most one member, by the definition of noncopying sequence (Def. 37, p. 55). Since $N_0 \subseteq \underline{imr}A_0$, $\hat{r}[A_0,(N_0)]M_1$ also has at most one member, so the first step $B_0 \xrightarrow[\hat{r}[A_0,(N_0)]M_1]{} B_1$ in (B_i) satisfies Definition 26. Since r is innermost preserving (Def. 35, p. 52), for all i, $N_i \subseteq \underline{imr}A_i$.

So commutativity guarantees that every subsequent step $B_{j-1} \xrightarrow[\hat{r}[A_{j-1},(N_{j-1})]M_j]{} B_j$ also satisfies Definition 37. So $B_0 \xrightarrow[\hat{r}[A_0,(N_0)]M_1]{} B_1 \xrightarrow[\hat{r}[A_1,(N_1)]M_2]{} \ldots$ is a noncopying sequence.

We need only show that (B_i) contains at most one more empty reduction than $A_0 \xrightarrow[M_1]{} A_1 \xrightarrow[M_2]{} \ldots$.

Suppose $M_{i+1} \neq \phi$ but $\hat{r}[A_i,(N_i)]M_{i+1} = \phi$. Then, since $N_i \subseteq \underline{imr}A_i$, $M_{i+1} \subseteq N_i$. So, by clause (2) of Definition 37, $M_{i+1} = N_i \neq \phi$.

Now, $N_{i+1} = \hat{r}[A_i,(M_{i+1})]N_i = \phi$. Therefore, there is at most one i for which $M_{i+1} \neq \phi$ and $\hat{r}[A_i,(N_i)]M_{i+1} = \phi$, that is, $\underline{cost}(B_i) \geq \underline{cost}(A_i)-1$.
□

3. Strictly Innermost Sequences are Most Costly

Strictly innermost sequences are infinite whenever possible, so it seems reasonable that they may be very costly even if finite. By the second clause of the definition of an extended equivalent address map (Def. 37, p. 55), equivalent address sets in strictly innermost noncopying reduction sequences are all singletons. Therefore, the strictly innermost noncopying sequences are exactly the sequences $A_0 \xrightarrow[M_1]{} A_1 \xrightarrow[M_2]{} \ldots$ in which each M_i is empty or contains exactly one innermost redex.

Theorem 12 (Cost of Strictly Innermost Sequences)

Let $<\Sigma,F,\rightarrow,A>$ be a SRS which commutes with respect to an innermost preserving residual map r.

Let $A_0 \xrightarrow[M_1]{} A_2 \xrightarrow[M_2]{} \cdots \xrightarrow[M_n]{} A_n \in I^S$ be a noncopying sequence, with A_n in normal form. Let (B_i) be any noncopying sequence with $B_0=A_0$. Then $\underline{cost}(B_i) \le \underline{cost}(A_i)$.

Proof By induction on $\underline{cost}(A_i)$.

Basis: If $\underline{cost}(A_i) = 0$, then A_0 is in normal form, and no nontrivial reduction sequence may start with A_0.

Induction step: Assume that the theorem holds for less costly sequences. Let M_j be the first nonempty redex set in $A_0 \xrightarrow[M_1]{} A_1 \xrightarrow[M_2]{} \cdots \xrightarrow[M_n]{} A_n$.
$A_0 \xrightarrow[M_j]{} A_j$, and $A_j \xrightarrow[M_{j+1}]{} \cdots \xrightarrow[M_n]{} A_n$ is a less costly sequence.
Let $N_0=M_j$, and define (N_i) and (C_i) by $(B_i) \xrightarrow[(N_i)]{} (C_i)$. Notice that $|N_0|=|M_j|= 1$. By Lemma 21 (p. 60), (C_i) is a noncopying sequence, and $\underline{cost}(C_i) \ge \underline{cost}(B_i)-1$. By induction hypothesis,
$\underline{cost}(C_i) \le \underline{cost}(A_j \xrightarrow[M_{j+1}]{} \cdots \xrightarrow[M_n]{} A_n) = \underline{cost}(A_i)-1$.
So, $\underline{cost}(B_i) \le \underline{cost}(A_i)$. □

4. Some Strictly Outermost Sequence is Optimal

Exactly characterizing optimal noncopying reduction sequences without additional constraints on SRSs may be very difficult. But, appealing only to commutativity and the outer property, Theorem 13 below at least restricts attention to reduction sequences which always reduce some outermost redex.

Definition 39

A noncopying reduction sequence $A_0 \xrightarrow[M_1]{} A_1 \xrightarrow[M_2]{} \cdots$ is a strictly outermost sequence iff, for all i, $M_{i+1} \cap \underline{omr}A_i \ne \phi$.
$O^S = \{(A_i) \mid (A_i)$ is a strictly outermost sequence$\}$

Note that to specify precisely a strictly outermost noncopying sequence, we need only choose one outermost redex, x, at each step, and reduce the entire equivalent address set containing x. The "delay-rule" of Vuillemin [Vu74] (3.1, p. 232) generates a strictly outermost non-copying sequence. In order to compare strictly outermost noncopying sequences to other noncopying sequences, we will need an inverse of the equivalent address map.

Definition 40

$\hat{e}^{-1}[A,(M_1,\ldots,M_n)]N =$
$\qquad \{x \mid x$ is a redex in A and $\hat{e}[A,(M_1,\ldots,M_n)]\{x\} \subseteq N\}$

Now, we may commute outermost and nonoutermost reduction steps in a noncopying sequence.

Lemma 22

Let <Σ,F,→,A> be outer and commutative with respect to r.
Let $A_0 \xrightarrow[M_1]{} A_1 \xrightarrow[M_2]{} \cdots$ be a noncopying reduction sequence, and let $M_i \cap \underline{\text{omr}}A_{i-1} = \phi$ but $M_{i+1} \cap \underline{\text{omr}}A_i \neq \phi$. Then the sequence

$$A_0 \xrightarrow[M_1]{} \cdots A_{i-1} \xrightarrow[\hat{e}^{-1}[A_{i-1},(M_i)]M_{i+1}]{} A'_i \xrightarrow[\hat{r}[A_{i-1},(\hat{e}^{-1}[A_{i-1},(M_i)]M_{i+1})]M_i]{} A_{i+1} \xrightarrow[M_{i+2}]{} \cdots$$

is a noncopying sequence which costs no more than the original sequence $A_0 \xrightarrow[M_1]{} \cdots A_{i-1} \xrightarrow[M_i]{} A_i \xrightarrow[M_{i+1}]{} A_{i+1} \xrightarrow[M_{i+2}]{} \cdots$ and $\hat{e}^{-1}[A_{i-1},(M_i)]M_{i+1} \cap \underline{\text{omr}}A_{i-1} \neq \phi$.

Proof We must complete the diagram of Figure 28, where the right-hand path $A_{i-1} \xrightarrow[M_i]{} A_i \xrightarrow[M_{i+1}]{} A_{i+1}$ is given.

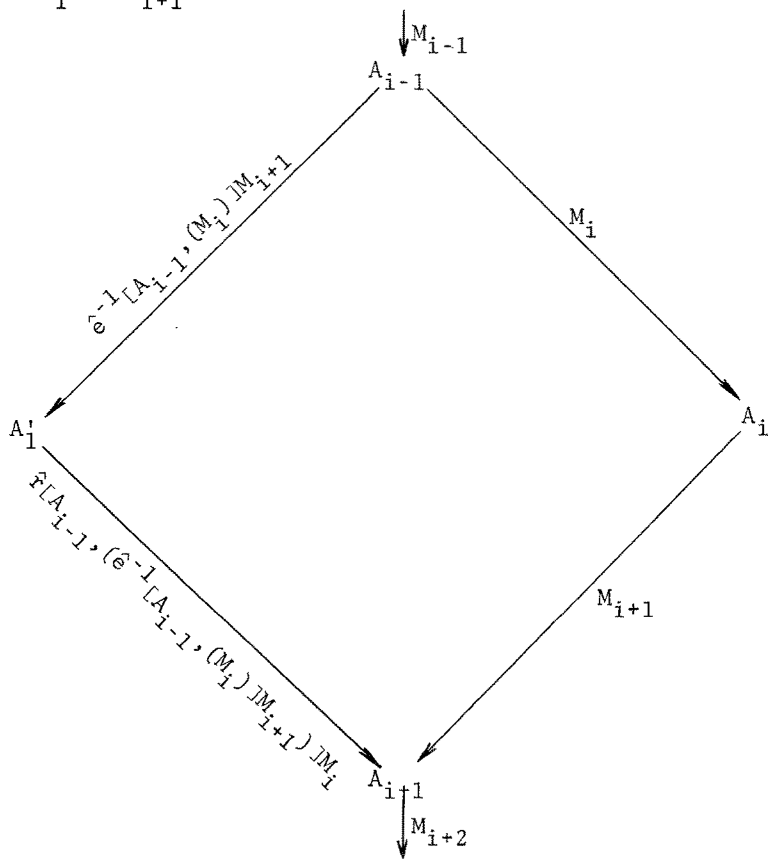


Figure 28

Since $M_i \cap \underline{omr}A_{i-1} = \phi$, $\underline{omr}A_{i-1} = \underline{omr}A_i$ by Lemma 14. A short argument establishes that $M_{i+1} \cap M_i = \phi$, and therefore $\hat{e}^{-1}[A_{i-1},(M_i)]M_{i+1}$ are equivalent redexes in A_i and $\hat{r}[A_{i-1},(M_i)](\hat{e}^{-1}[A_{i-1},(M_i)]M_{i+1}) = M_{i+1}$. So

$A_{i-1} \xrightarrow[\hat{e}^{-1}[A_{i-1},(M_i)]M_{i+1}]{} A_i' \xrightarrow[\hat{r}[A_{i-1},(\hat{e}^{-1}[A_{i-1},(M_i)]M_{i+1})]M_i]{} A_{i+1}$ by Lemma 12.

The new sequence is noncopying by Lemma 20. □

<u>Theorem 13</u> (Optimal Sequence Theorem)

Let $<\Sigma,F,\rightarrow,A>$ be outer and commutative with respect to r. Assume A_0 has normal form D. Then there is a strictly outermost noncopying sequence from A_0 to D which is optimal (costs no more than any other noncopying sequence from A_0 to D).

<u>Proof</u> Let $A_0 \xrightarrow[M_1]{} A_1 \xrightarrow[M_2]{} \cdots A_{n-1} \xrightarrow[M_n]{} A_n$ be any optimal noncopying sequence ending in normal form A_n. We will use Lemma 22 to transform (A_i) into a strictly outermost noncopying sequence without increasing the cost.

(*) First eliminate all empty reductions.

$M_n = \underline{omr}A_{n-1}$, else, by closure (Def. 25, p. 30) A_n would contain a redex.

Let M_i be the last redex set (if any) such that $M_i \cap \underline{omr}A_{i-1} = \phi$. $i<n$, so $M_{i+1} \cap \underline{omr}A_{i+1} = \phi$, and by Lemma 22 we may commute M_i and M_{i+1} producing another noncopying sequence with no increase in cost. Repeat this process from (*) until a strictly outermost sequence is obtained. □

5. <u>Finding an Optimal Sequence Using a Dominance Ordering</u>

Consider E of Example 6 (p. 18). The expression Cond(p(s(0)),F(0,0),F(s(0),0)) has three outermost redexes, (1), (2) and (3), any of which might be chosen by a strictly outermost sequence. An intuitive understanding of the conditional function Cond suggests that (1) be reduced first, since the value of the argument at (1) determines which of the other two arguments is relevant. Notice that reduction of (1) creates a redex Cond(0,F(0,0),F(s(0),0)) at the outermost address (), while reduction of (2) and (3) cannot create such a redex. The address (1) <u>dominates</u> its brothers (2) and (3) in the sense that only a reduction at (1) may create a redex at ().

<u>Definition 41</u>

A <u>dominance ordering</u> is a function $d:F\rightarrow P^*\times P^*$ such that, for all $A\in F$,

(1) dA is a partial ordering of domainA,

(2) x anc y => x(dA)y for x,y $\in$ domainA,

(3) x(dA)y(dA)z $\wedge$ $A \xrightarrow[z]{} B$ => x(dB)y.

x(dA)y is read "x dominates y in A".

A SRS $<\Sigma, F, \rightarrow, A>$ is d-outer iff the following holds:
Let A and B be in F, and let x, y and z be in domainA with x(dA)y(dA)z, y$\neq$z. Assume x is not a redex in A and y and z are redexes in A, and $A \xrightarrow[z]{} B$. Then x is not a redex in B.

Definition 41 generalizes Definition 32 (p. 46). Note that the ordering defined by x(dA)y <=> x anc y is a dominance ordering. Another useful dominance ordering is preorder.

Definition 42

Preorder is the dominance ordering defined by
x(dA)y iff x anc y
or $\exists_{z,i,j}$ z$\cdot$(i) anc x $\wedge$ z$\cdot$(j) anc y $\wedge$ i<j

When d is preorder, strictly d-outermost reduction is called leftmost outermost reduction.

The SRS E of Example 6 (p. 18) is d-outer for preorder d, as is that of Example 18 (p. 56).

All the results of Chapters V and VI generalize to arbitrary dominance orderings. The following definition generalizes Definitions 29-31 (pp. 44, 45).

Definition 43

x is a d-outermost redex in A iff x is a redex in A and
$\neg\exists_y$y(dA)x $\wedge$ y$\neq$x $\wedge$ y is a redex in A.

$\text{omr}_d A$ = {x| x is a d-outermost redex in A}.

Let $A_0 \xrightarrow[M_1]{} A_1 \xrightarrow[M_2]{} \ldots$ be a reduction sequence with x $\in$ $\text{omr}_d A_j$. x in $\text{omr}_d A_j$ is eliminated as a d-outermost redex in A_k the first time that (1) $x \in M_k$ or (2) $x \notin \text{omr}_d A_k$.

The reduction sequence (A_i) is an eventually d-outermost reduction sequence iff

$\forall_{j \geq 0, x \in \text{omr}_d A_j} \exists_{k>j}$ x is eliminated in A_k.

O_d^e = {(A_i)| (A_i) is an eventually d-outermost reduction sequence}.

Note that $\text{omr}_d A \subseteq \text{omr}A$ and $O^e \subseteq O_d^e$.

Now we may generalize the Eventually Outermost Sequence Theorem.

Theorem 14

Let <Σ,F,→,A> be closed and d-outer.
Let $A_0 \xrightarrow[M_1]{} A_1 \xrightarrow[M_2]{} \ldots \in \mathcal{O}_d^e$ and let A_0 have normal form D.
Then $\exists_\ell A_\ell = D$.

Proof Analogous to proof of Theorem 10 (p. 50). □

The following definition generalizes Definition 39 (p. 61).

Definition 44

A noncopying reduction sequence $A_0 \xrightarrow[M_1]{} A_1 \xrightarrow[M_2]{} \ldots$ is a strictly d-outermost sequence iff, for all i, $M_{i+1} \cap \underline{omr}_d A_i \neq \phi$.
$\mathcal{O}_d^s = \{(A_i) \mid (A_i)$ is a strictly d-outermost sequence$\}$.

Note that $\mathcal{O}_d^s \subseteq \mathcal{O}^s$.

Finally, we generalize the Optimal Sequence Theorem.

Theorem 15

Let <Σ,F,→,A> be d-outer and commutative with respect to r. Assume A_0 has normal form D. Then there is a strictly d-outermost noncopying sequence from A_0 to D which is optimal.

Proof Analogous to proof of Theorem 13 (p. 63). □

Notice that if dA is a total ordering for all A∈F (e.g., preorder is a total ordering), then a strictly d-outermost noncopying sequence is uniquely determined by the starting expression A_0, and $\mathcal{O}_d^s \subseteq \mathcal{O}_d^e$. So, if a SRS E is d-outer for an easily computable total dominance ordering d, we may easily generate optimal noncopying reduction sequences for E. If E is d-outer but d is not total, a reasonable ad hoc strategy is to generate sequences in $\mathcal{O}_d^s \cap \mathcal{O}_d^e$, since this set is terminating and contains the optimal sequences.