Notre Dame Journal of Formal Logic
Volume IX, Number 3, July 1968

MODIFIED STRONG REDUCTION IN COMBINATORY LOGIC

KENNETH LOEWEN

Introduction. This paper introduces a modified definition of strong reduction in combinatory logic. A number of simple properties of reductions are developed. Then the modified definition is shown to be equivalent to the original definition in the sense that any modified reduction from an ob X to an ob Y can be replaced by a reduction in the original sense from the same X to the same Y.

A Modified Definition of Strong Reduction in Combinatory Logic. Strong Reduction in Combinatory Logic was introduced in Curry and Feys *Combinatory Logic* [1]. Prior to this time they used only the weak reduction rules

$$(\mathsf{I}) \qquad \mathsf{I}X \geq X;$$

$$(\mathsf{K}) \qquad \mathsf{K}XY \geq X;$$

$$(\mathsf{S}) \qquad \mathsf{S}XYZ \geq XZ(YZ);$$

together with rules of combinators which can be defined in terms of these. Here X, Y, and Z are arbitrary combinations of elements of the system. To obtain a closer analog with some of the related systems of lambda conversion [2] Curry and Feys introduced strong reduction. In this definition of strong reduction bound variables are introduced. Expressions containing these bound variables are called lambda expressions since a lambda is used to indicate the variables which are bound.

In the definition of strong reduction (given in detail below) lambda expressions were not allowed in the interior of a reduction except for steps which removed them. The purpose of this paper is to show that these restrictions can be relaxed to a great extent. In this we get an answer to a problem listed as unsolved in Curry and Feys [1a]. The modification was suggested by steps used in proving a standardization theorem in the writer's doctoral dissertation [3]. This modification allows essential simplifications in that proof.

Strong reduction is formulated in Curry and Feys as a formal system whose entities are called obs. Actually several different systems are used.

Received August 18, 1966

In the first formulation--the system $\mathcal{H}$--there are no lambda expressions. Its obs are called $\mathcal{H}$-obs. In a modified system (called $\mathcal{J}$) lambda expressions are introduced and in this system we have $\mathcal{J}$-obs.

A reduction consists of a replacement of a component (called a redex) of an ob of the system under consideration by another component called the contractum of the redex. The replacements which are allowed are given by the rules of the system.

The use of parentheses in what follows is simplified by assuming associativity to the left and the use of dots as parentheses in connection with lambda expressions. The symbol $\succ$ is used as a binary infix to denote a strong reduction.

Curry and Feys define strong reduction as a sequence of steps of the following three types:

Type I. Replacement according to one of the following three rules where X, Y, and Z represent arbitrary $\mathcal{H}$-obs (i.e. obs without any lambda expressions inside them):

	Redex	Contractum
(I)	$\mathsf{I}X$	X.
(K)	$\mathsf{K}XY$	X.
(S)	$\mathsf{S}XYZ$	$XZ(YZ)$.

Type II. Replacement of a component U which is an $\mathcal{H}$-ob by $\lambda x.Ux$, where x is a variable which does not occur in U.

Type III. Replacement of a component $\lambda x.\overline{\mathfrak{X}}$ by $X = [x]\overline{\mathfrak{X}}$, where $\overline{\mathfrak{X}}$ is an $\mathcal{H}$-ob and $[x]\overline{\mathfrak{X}}$ is defined by the algorithm:

(a) $[x]X \equiv \mathsf{K}X$ if $\overline{\mathfrak{X}}$ is X and X does not contain x.

(b) $[x]x \equiv \mathsf{I}$ if $\overline{\mathfrak{X}}$ is x.

(c) $[x]\overline{\mathfrak{Y}\mathfrak{Z}} \equiv \mathsf{S}([x]Y)([x]Z)$ if $\overline{\mathfrak{X}}$ is $\overline{\mathfrak{Y}\mathfrak{Z}}$ and none of the previous cases apply.

The relation generated by these steps is reflexive, transitive and both right and left monotonic [1b].

In the expressions U and V there may be indeterminates but x may not appear. In practice Type II steps are modified so as to include one or more Type II steps as given in the definition followed immediately by a Type I step to eliminate the initial combinator. If this were not done, the Type II step could be postponed.

Modified Definition. The following modification of strong reduction is proposed here: In steps of Type I and II drop the requirement that the arguments of the combinators be $\mathcal{H}$-obs and allow them to be $\mathcal{J}$-obs. In other words, we will allow lambda expressions within U, X, Y and Z. This same modification cannot be extended to Type III steps.

We can summarize the definition of this modified strong reduction in tabular form. In the listings for Type II steps the steps actually consist of one or more Type II steps followed by one Type I step to eliminate the initial combinator.

Type	Redex	Contractum
Ia	$\mathsf{K}XY$	X;
Ib	$\mathsf{S}XYZ$	$XY(YZ)$;
Ic	$\mathsf{I}X$	X;
IIa	$\mathsf{K}U$	$\lambda x.U$;
IIb	$\mathsf{S}UV$	$\lambda x.\,Ux(Vx)$;
IIc	I	$\lambda x.x$;
IId	K	$\lambda xy.x$;
IIe	$\mathsf{S}U$	$\lambda xy.Uy(xy)$;
IIf	S	$\lambda xyz.xz(yz)$;
III	$\lambda x.V$	$[x]V$.

Here U, X, Y and Z may contain lambda expressions, but V may not.

From these definitions we can make several observations about redexes. A particular redex ceases to be a redex when it is contracted. In a few instances we will refer to the contractum of a given redex as its trace. This is done only when the contractum is identifiable as an entity.

A Type III redex also ceases to be a redex when a Type II step takes place in its interior. This comes from the requirement that all components of Type III redexes be $\mathscr{H}$-obs. When this happens, it will be convenient to call the resulting expression in quasi-redex. Thus a quasi-redex is an expression satisfying the requirements for a Type III redex except that it may have lambda expressions in its interior. Since there is no restriction on the form of V in a Type III redex other than that it be an $\mathscr{H}$-ob, a quasi-redex can be converted into a redex by appropriate Type III steps.

Type II and III steps can be reversed under certain conditions as given in the following two lemmas:

Lemma 1. *If $X = [x]\bar{\mathfrak{X}}$, then $\lambda x.Xx \succ\!\!- \lambda x.\bar{\mathfrak{X}}$ by Type I steps only. In other words the contractum of a Type III step may be reversed to the original redex by a single Type II step followed by Type I steps.*

Proof: This is a restatement of the conclusion of the discussion of the first part of Curry and Feys Section 6A2 [1c].

Lemma 2. *The contraction of a Type II redex P may be reversed provided there are no intervening steps interior to the contractum P.*

Proof: If the arguments of the initial combinator of P contain no lambda expressions, this follows directly from the algorithm for Type III steps.

If one or more of the arguments contain lambda expressions in their interior, these must be removed by Type III steps. After the reversal of P, the interior lambda expressions may be reintroduced by Lemma 1.

Note that the reversals provided by Lemmas 1 and 2 are both valid strong reductions.

We shall call the initial combinator of a Type I or II redex the *head* of that redex. In the case of a Type III redex the head will be the λx. If two different redexes have the same occurence of a combinator as their heads (they are necessarily Type I or II since two Type III redexes with the same occurence of λx as their heads are identical), we will call them competing redexes. If one of two competing redexes is contracted, the other one is no longer a redex.

A redex can also be cancelled as a whole by an application of rule (K).

If a redex of Type I or II is interior to a Type III redex and the Type III redex is contracted, the parts of the contractum arising from the Type I or II redex need not form a redex.

We now introduce the concept of *residual.* If a given expression contains two redexes P and Q, and if P is contracted, then we would like to associate certain components in the resulting expression with the redex Q. These associated components will be known as residuals. The definition actually includes the case where Q is a quasi-redex of Type III.

Definition. Let P be a redex and Q a (quasi-)redex in M and let the contractum of P be L. Let the contraction of P in M reduce M to N; then the residuals of Q are those components of N defined as follows:

Case 1. P is the same as Q. Then Q has no residual.

Case 2. P and Q do not overlap. N is obtained from M by replacing the component P by L. Every component not overlapping P in M will have a homologous component in N. The component in N which is homologous to Q is the residual of Q.

Case 3. P is part of Q not including the head of Q. The component Q' of N arising from the (quasi-)redex Q of M by replacing the subcomponent P of Q by L is the residual of Q.

Case 4. Q is a part of P not including the head of P. If P is a Type I or II redex, and p is the initial combinator of P, then P is of the form $pX_1 \ldots X_m$ and Q is in some X_i. Arguments are not changed except for multiplicity in a contraction. Hence components congruent to Q and arising from Q in M will appear in N with a possible change in multiplicity. These components occurring in N congruent to Q and arising from Q in M are the residuals of Q.

If P is a Type III redex, we consider two possibilities. If Q does not contain x and appears as a subcomponent in a component of M of the form $M_1 x$ or simply M_1 where M_1 does not contain x, then N will contain an instance of the component M_1 and hence its subcomponent Q which is then the residual of Q in M. If Q does not enter into M in the way described, then N will contain $[x]Q$ and this will be the residual of Q. $[x]Q$ need not be a redex of the same type as Q.

Case 5. P and Q have the same head, but are not the same redex. Here Q has no residual.

This completes the definition of a residual of a (quasi-) redex after a single contraction. In a sequence of contractions we need only add that any residual of a residual is again a residual.

Note that if P is a Type I or II redex and Q is a (quasi-) redex, then if Q has more than one residual after the contraction of P, the several residual redexes will be disjoint. The only case in which there can be an increase in multiplicity is case 4 with P a redex of Type Ib. In none of the other cases is there an increase in the multiplicity of argument places except for those filled by indeterminates. Arguments are not changed in a reduction except for possible changes in multiplicity. In this case P is of the form $\mathsf{S}XYZ$ which reduces to $XZ(YZ)$. If Q occurs once in Z before P is contracted, then there will be a residual of Q in each occurence of Z after the contraction. Clearly these must be disjoint (although another redex could contain both instances of Z and hence Q). If Z contains several residuals of Q before contraction these must have come from earlier steps of this type and hence will be disjoint. If X or Y contain residuals of Q, these must also be disjoint from any which occur in Z before the reduction and hence will be disjoint after the reduction since X and Y are not changed by the reduction.

In order to prove the equivalence of strong reduction and modified strong reduction we observe first of all that any strong reduction is also a modified strong reduction. Hence we need only show that any modified strong reduction can be replaced by a valid strong reduction. The definition of Type III steps is the same for both reductions so we need consider only Type I and II reductions.

If we have a modified strong reduction from M to N we proceed to construct a strong reduction from M to N. This proceeds by an induction on the structure of the reduction. Let the i-th step in the reduction be the contraction of a redex P taking M_{i-1} into M_i. If P is a Type III redex the step in the new reduction is the same as the old reduction. If P is a Type I or II reduction and P does not contain any lambda expressions the new step is again the same as the old one.

Finally suppose P is a Type I or II redex containing a Type III (quasi-) redex Q_1. Q can be reduced by Type III steps to Q' which contains no lambdas. Similarly for any other Type III redex Q_i contained in P. Then P can be reduced as a strong reduction. Each Q_i' will then have one or more residuals in the resulting expression.

By Lemma 1 if X does not contain x, then $[x]Xx$ reduces to X by Type I steps only. Hence if we apply one Type II step to each residual of each Q_i' and then follow these by appropriate Type I steps we will obtain an expression which has a redex congruent to Q_i as a residual of Q_i, but this is just M_i. Since all of these steps are valid strong reduction steps we have constructed a strong reduction from M_{i-1} to M_i. The number of steps following M has not been affected and we can complete the induction.

A counter example to show that the restriction of $\mathcal{U}$-obs cannot be relaxed in the case of Type III steps is as follows:

$$\lambda xy.x = \lambda x([y]x) = \lambda x\,\mathsf{K}x = \mathsf{K}.$$

If the other order were allowed we would get

$$\lambda xy.x = [x](\lambda y.x) = \lambda y\mathsf{I} = [y]\mathsf{I} = \mathsf{KI}.$$

But K and KI do not have the same effect as is shown by

$$\mathsf{K}XY \geq X;$$

$$\mathsf{KI}XY \geq \mathsf{I}\,Y \geq Y.$$

REFERENCES

[1] Curry, H. B., and Robert Feys, ***Combinatory Logic Vol. 1*** (Studies in Logic and Foundations of Mathematics Series) North Holland Publishing Co., Amsterdam, 1958.

[1a] The first problem listed in section 6FS on page 236.

[1b] See Curry and Feys, Theorem 2D1.

[1c] Pages 188, 189. See also the proof of Theorem 6D2, page 207.

[2] For these see A. Church, ***The Calculi of Lambda Conversion,*** Annals of Mathematics Studies 6, Princeton, N. J.; 1941, 2nd. edition, 1951.

[3] Loewen, Kenneth: ***A Study of Strong Reduction in Combinatory Logic,*** a Ph.D. thesis at The Pennsylvania State University, University Park, Pennsylvania, 1962. Written under the direction of H. B. Curry.

University of Oklahoma
Norman, Oklahoma