“Delayability” in Proofs of Strong Normalizability in the Typed Lambda Calculus

Michael Karr*
Software Options, Inc.
22 Hilliard Street
Cambridge, Mass 02138

1. Introduction

We consider here the interaction of certain reductions with the β-reduction $(\lambda x.M)N \to [N/x]M$ (substitute N for occurrences of x in M). Given a term in the typed λ-calculus, any sequence of β-reductions must eventually terminate; this is referred to as the *strong normalizability* of the term, or in this paper, simply *normalizability*. It is often useful to consider other reductions in addition to the β-reduction. The most classical example is

$\lambda x.Mx \to M$ when x is not free in M (η-reduction)

Other reductions arise when considering terms extended with the constants left, right and pair, and the reductions:

left (pair x y) $\to x$ (L-reduction)

right (pair x y) $\to y$ (R-reduction)

pair (left z) (right z) $\to z$ (P-reduction)

Most works on the typed lambda calculus consider normalizability for only the β- and η-reductions, for example, [1] and [6]. Using a standard technique for proving strong normalizability [8], it is not difficult to show that applying η-, L-, and R- reductions in addition to β-reduction still leaves all terms strongly normalizable. It is more difficult to show this when P-reductions are added to the set [2], [4]. It is also possible to handle these reductions by generalization of the notion of type [7]. A different approach to proving normalizability has been applied to the L- and R-reductions [3]. The goal here is to have a general theory that allows us to adjoin new reductions to the β-reduction, and by simple structural examination of the new reductions, ascertain that terms remain normalizable in the new system. To do this, we have developed a new strategy for proving strong normalizability, which as a special case yields an independent proof of normalizability of terms under β-reduction. While we use the pairing reductions to exemplify the general results, the emphasis is on clarifying the properties of reductions that allow the proof of normalizability.

In the next section, we develop a general technique for proving that a union of well-founded relations is well-founded, given original relations that are well-founded and whose interaction obeys certain conditions. The third section gives a precise meaning to the term “reduction”, and provides results that are helpful in proving the conditions required for the well-foundedness results. In the following section, we show that a broad class of reductions, including the union of the β-, L-, and R-reductions, has the normalizability property. The final section considers reductions having other properties; this class includes the η- and P-reductions.

---

*Part of this work was done at Harvard University, where the author was supported by an IBM graduate fellowship.

2. Well-foundedness of a union of relations

2.1. Delayability

Let $\rightarrow_i$ denote a relation, $\twoheadrightarrow_i$ its reflexive transitive closure, and $\underset{\neq}{\twoheadrightarrow}_i$ its transitive closure. We shall investigate the consequences of the following condition on relations.

Definition A relation $\rightarrow_0$ *is delayable past* $\rightarrow_1$ via $\rightarrow_2 \overset{\text{def}}{\Leftrightarrow}$

$$M \rightarrow_0 M' \rightarrow_1 M'' \Rightarrow \exists N', N'': M \underset{\neq}{\twoheadrightarrow}_1 N' \twoheadrightarrow_0 N'' \text{ and } M'' \twoheadrightarrow_2 N''$$

□

The following definition is standard.

Definition A pair of relations $\rightarrow_0, \rightarrow_1$ is *commutative* $\overset{\text{def}}{\Leftrightarrow}$ $M \rightarrow_i M_i$, $i = 0$ and $1 \Rightarrow \exists M_2 : M_i \rightarrow_{1-i} M_2$

□

Lemma 1 Suppose that

$\rightarrow_0$ is delayable past $\rightarrow_1$ via $\rightarrow_2$,

$\twoheadrightarrow_i, \twoheadrightarrow_2$ is commutative for both $i = 0$ and 1,

$\underset{\neq}{\twoheadrightarrow}_k, \twoheadrightarrow_2$ is commutative for either $k = 0$ or 1,

$\rightarrow_i$ is well-founded, $i = 0$ and 1.

Then $\rightarrow_{01} \triangleq \rightarrow_0 \cup \rightarrow_1$ is well-founded.

Proof Suppose we have an infinite path in $\rightarrow_{01}$. Because the $\rightarrow_i$ are individually well-founded, there are an infinity of alternations between $\rightarrow_0$ steps and $\rightarrow_1$ steps. We may write the path in the form:

$$M_0 \twoheadrightarrow_1 M_1 \underset{\neq}{\twoheadrightarrow}_0 M_2 \underset{\neq}{\twoheadrightarrow}_1 M_3 \underset{\neq}{\twoheadrightarrow}_0 \cdots$$

The crux of the proof is that we can shorten $M_1 \underset{\neq}{\twoheadrightarrow}_0 M_2$ by one step, and obtain another infinite sequence. When there is only one $\rightarrow_0$ step between M_1 and M_2, and it is shortened, we have extended $M_0 \twoheadrightarrow_1 M_1$ by one more $\rightarrow_1$ step. The ability to do this indefinitely denies the well-foundedness of $\rightarrow_1$.

Look at the term N_1 just preceding M_2 in the sequence, and the term N_2 just following M_2. Using delayability:

$$N_1 \rightarrow_0 M_2 \rightarrow_1 N_2 \Rightarrow \exists P_1, P_2: N_1 \underset{\neq}{\twoheadrightarrow}_1 P_1 \twoheadrightarrow_0 P_2 \text{ and } N_2 \twoheadrightarrow_2 P_2$$

Then by using the commutativity conditions starting with $N_2 \rightarrow_1 M_3$ and $N_2 \rightarrow_1 P_2$, we can get a new infinite sequence by extending the "ladder" one step at a time:

$$\begin{array}{ccccccccccccccc}
M_0 & \twoheadrightarrow_1 & M_1 & \twoheadrightarrow_0 & N_1 & \rightarrow_0 & M_2 & \rightarrow_1 & N_2 & \twoheadrightarrow_1 & M_3 & \underset{\neq}{\twoheadrightarrow}_0 & M_4 & \underset{\neq}{\twoheadrightarrow}_1 & \cdots \\
\parallel & & \parallel & & \parallel & & & & \downarrow & & \downarrow & & \downarrow & & \\
M_0 & \twoheadrightarrow_1 & M_1 & \twoheadrightarrow_0 & N_1 & \underset{\neq}{\twoheadrightarrow}_1 & P_1 & \twoheadrightarrow_0 & P_2 & \twoheadrightarrow_1 & P_3 & \twoheadrightarrow_0 & P_4 & \twoheadrightarrow_1 & \cdots
\end{array}$$

where the relations $\downarrow$ are $\twoheadrightarrow_2$. By positive commutativity, $P_{2i+k-1} \neq P_{2i+k}$, and so the new sequence is indeed not eventually constant. As we claimed, either the first set of $\rightarrow_0$ steps has been shortened, or the first set of $\rightarrow_1$ steps has been lengthened (perhaps from zero to one).

□

Corollary 1 Suppose that $\rightarrow_0$ is delayable past $\rightarrow_1$ via $=$, and that $\rightarrow_i$ is well-founded for $i = 0$ and 1. Then $\rightarrow_0 \cup \rightarrow_1$ is well-founded.

Proof Apply the Theorem, because $\rightarrow$, $=$ is commutative for any relation $\rightarrow$.

□

The following result is useful in establishing the delayability properties of a union of relations (simple proofs are left to the reader).

Proposition 2.1 If $\to_0$ and $\to_1$ are delayable past $\to_2$ via $\to_3$ for $i = 0$ and 1, then $\to_0 \cup \to_1$ is delayable past $\to_2$ via $\to_3$.

□

2.2. Weak Commutativity

When $\to_1$ is not equality, the commutativity of $\underset{\neq}{\twoheadrightarrow}_0, \twoheadrightarrow_1$ may be proved by using a property of the relations $\to_0, \to_1$.

Definition An (ordered) pair of relations $\to_0, \to_1$ is *positive weak commutative* $\overset{\text{def}}{\Leftrightarrow}$

$$M \to_i M_i,\ i = 0 \text{ and } 1 \Rightarrow \exists M_2 : M_1 \underset{\neq}{\twoheadrightarrow}_0 M_2 \text{ and } M_0 \twoheadrightarrow_1 M_2$$

(*Weak commutativity* is the same except that $\underset{\neq}{\twoheadrightarrow}_0$ is replaced by $\twoheadrightarrow_0$.)

□

In addition to this property that involves the pair, it is convenient to work with a relation that has the following property.

Definition A relation $\to$ is *bounded* $\overset{\text{def}}{\Leftrightarrow}$ $\exists \nu\ \forall M$: any $\to$ path from M has at most $\nu(M)$ steps.

□

Lemma 2 Suppose that

$\to_0, \to_1$ is positive weak commutative.

$\to_0$ is bounded.

Then $\underset{\neq}{\twoheadrightarrow}_0, \twoheadrightarrow_1$ is commutative.

Proof Proofs omitted in this paper that appear in the full paper [5] are indicated by ○.

○

Corollary 2 If $\to,\to$ is weak commutative and $\to$ is bounded, then $\twoheadrightarrow,\twoheadrightarrow$ is commutative.

○

These results show that the issue of commutativity of a transitive closure reduces to the issue of (positive) weak commutativity and boundedness. Proofs of (positive) weak commutativity are often aided by:

Proposition 2.2 If $\to_0,\to_1$ and $\to_0,\to_2$ are (positive) weak commutative, then $\to_0,\to_1 \cup \to_2$ is (positive) weak commutative. If $\to_0,\to_2$ and $\to_1,\to_2$ are (positive) weak commutative, then $\to_0 \cup \to_1,\to_2$ is (positive) weak commutative.

□

Boundedness is often proved using the following property of a relation.

Definition A relation $\to$ *has finite degree* $\overset{\text{def}}{\Leftrightarrow}$ $\forall M$, there are a finite number of N such that $M \to N$.

□

Proposition 2.3 If a relation is well-founded and has finite degree, then it is bounded.

Proof This is equivalent to the well-known result that an infinite tree with finite branching has an infinite path.

□

Proposition 2.4 If $\to_0$ and $\to_1$ have finite degree, then so does $\to_0 \cup \to_1$.

□

2.3. Summary

The results of the rest of this paper will be concerned with proving the hypotheses of the result below, for various families of relations.

Definition A family $\{ \to_i \}_{i \in I}$ of relations *has finite application* $\overset{\text{def}}{\Leftrightarrow}$

$\forall M \; \exists$ a finite set I_M : in any path beginning at M, each step $\to_i$ has $i \in I_M$.

□

Proposition 2.5 If $\{ \to_i \}_{i \in I}$ has finite application and each $\to_i$ has finite degree, then $\to_\infty$ has finite degree.

□

Lemma 3 Let $\{ \to_i \}_{i \in I}$ be a family of relations over a well-ordered index set I, and let $\to_\infty \triangleq \bigcup_i \to_i$. Suppose, with cases a) and b):

$\{ \to_i \}_{i \in I}$ has finite application.

$\forall i : \to_i$ is well-founded.

$\forall i : \to_i$ has finite degree.

Case a) $\forall i < j : \to_i$ is delayable past $\to_j$ via $=$.

Case b) $\forall i < j : \to_i$ is delayable past $\to_j$ via $\to_j$.
$\forall i < j : \to_i, \to_j$ is positive weak commutative.
$\forall i : \to_i, \to_i$ is weak commutative.

Then $\to_\infty$ is well-founded.

○

Definition A relation $\to$ is *Church-Rosser* $\overset{\text{def}}{\Leftrightarrow}$ $\twoheadrightarrow, \twoheadrightarrow$ is commutative.

□

Corollary 3 Let $\{ \to_i \}_{i \in I}$ be as in Lemma 3, and suppose that in addition to previous hypotheses, we also assume that $\to_i, \to_j$ is weak commutative, $\forall \; i,j$. Then $\twoheadrightarrow_\infty$ is Church-Rosser.

○

Because of this result, we shall also be able to obtain general results for Church-Rosser properties of β-reduction adjoined by other reductions. Again, the emphasis will be on finding simple structural characteristics of the rules which guarantee the property.

3. Reductions

In this section, we will give a technical meaning to the term "reduction", as a type of relation on expressions in the typed λ-calculus. In the first subsection we discuss congruence closure; in particular, we are interested in proving delayability and weak commutativity properties of a relation from related properties on simpler relations. The next subsection discusses substitution and the third uses substitution and congruence closure to define reduction. The final subsection considers a situation in which a delayability condition is easily proved.

3.1. Congruence Closure

We shall work with relations $\to_0$ that have been defined beginning with a "top-level reduction" $\to^0$. Examples of top-level reduction are the β-, η-, and pairing reductions mentioned in the introduction. These relations may extended over expressions as follows:

Definition Given a relation $\to^0$ on expressions, the *congruence closure* $\to_0$ of $\to^0$ is defined inductively on terms, by $M \to_0 N \overset{\text{def}}{\Leftrightarrow}$

- $M \to^0 N$, or

- $M = M_1M_2$, $N = N_1N_2$, $M_i \to_0 N_i$ and $M_j = N_j$ for $i \neq j$, or

- $M = \lambda x.M_0$, $N = \lambda x.N_0$ and $M_0 \to_0 N_0$.

□

Proposition 3.1 The congruence closure of the (reflexive) transitive closure is the same as the (reflexive) transitive closure of the congruence closure.

□

Proposition 3.2 The congruence closure of $\to^0 \cup \to^1$ is $\to_0 \cup \to_1$.

□

By $\twoheadrightarrow^0$ we mean the reflexive transitive closure of $\to^0$; by $\twoheadrightarrow_0$, we mean the relation referred to by the above proposition. The notation $\to_{\bar{0}}$ means $\to_0 - \to^0$. As always, we are concerned with unions, delayability, commutativity, and finite degree.

Proposition 3.3 If $\to^0$ is delayable past $\to_1$ via $\to_2$ and $\to_{\bar{0}}$ is delayable past $\to^1$ via $\to_2$, then $\to_0$ is delayable past $\to_1$ via $\to_2$.

□

Proposition 3.4 If $\to^0,\to_1$ and $\to_{\bar{0}},\to^1$ are (positive) weak commutative, then $\to_0,\to_1$ is (positive) weak commutative.

□

Proposition 3.5 If $\to^0$ has finite degree, so does $\to_0$.

□

The proofs of all of these are by induction on terms.

3.2. Substitution

The definitions in this subsection will be used throughout this paper.

Definition A *substitution* s is a function mapping a finite set of variables to expressions, preserving type.

□

Thus, we may speak of Dom(s) and Range(s). It is often convenient to denote a function using specific variables and expressions. We use the notations

$$[M/x] \triangleq \{\langle x,M\rangle\},\ [M_i/x_i] \triangleq \{\langle x_i,M_i\rangle\}_i, \text{ etc.}$$

We also use the set operations on the functions, such as $\cup$ and $-$. We often need to remove a point from a domain, for which we use the notation $s{-}x \triangleq s{-}\{\langle x,sx\rangle\}$.

We denote the free variables of an expression M by FV(M), and extend this notation to FV(s), meaning the union of free variables of elements of Range(s). This is useful in defining how a substitution acts on an expresion:

Definition Inductively on terms:

$sc \triangleq c$	for constants
$sx \triangleq sx$ if $x \in \text{Dom}(s)$ x otherwise	for variables

$$s(M_1M_2) \triangleq (sM_1)(sM_2) \qquad \text{for applications}$$
$$s(\lambda x.M_0) \triangleq \lambda y.((s - x \cup [y/x])M_0) \qquad \text{for abstractions}$$

□

For variables $x \in \mathrm{Dom}(s)$, it is not clear whether sx refers to the result of applying s as a function or of doing the substitution. But the result is the same in both cases and the ambiguity is inconsequential. Note that substitution into abstractions is somewhat arbitrary. We view it as being well-defined only up to $=_\alpha$, the equivalence relation obtained by allowing renaming of bound variables. When we use "=" for expressions, we implicitly mean $=_\alpha$. Using these definitions and notation, top-level β-reduction is correctly defined by the rule:

$$(\lambda x.M)N \to^\beta [N/x]M$$

We shall view β-reduction in a somewhat different way in the next subsection.

3.3. Substitutive Closure

In this subsection, we formalize "reduction". We start with the following notion.

Definition A *reduction set* $\mathfrak{R}$ is a set of pairs of expressions $\langle P,Q \rangle$ where for every such pair in $\mathfrak{R}$:

$$\mathrm{type}(P) = \mathrm{type}(Q) \text{ and } \mathrm{FV}(P) \supseteq \mathrm{FV}(Q).$$

□

For reasons that will become apparent, when $\langle P,Q \rangle \in \mathfrak{R}$, P is called a *pattern* and Q is a *replacement*.

We may view $\mathfrak{R}$ as a relation on expressions. The first step is to get a new relation from it.

Definition Given a reduction set $\mathfrak{R}$, the *substitutive closure* of $\mathfrak{R}$, $\to^{\mathfrak{R}}$ is given by

$$M \to_{\mathfrak{R}} N \overset{\text{def}}{\Leftrightarrow} \exists \text{ a substitution } s \text{ and } \langle P,Q \rangle \in \mathfrak{R} : \mathrm{Dom}(s) = \mathrm{FV}(P),\ M = sP, \text{ and } N = sQ$$

□

Any $\to^{\mathfrak{R}}$ defined in this way is by definition a top-level reduction.

Reduction sets are often parameterized by type. For example, let:

$$\mathfrak{R}_\eta \triangleq \{ \langle \lambda x.qx,\ q \rangle : q \text{ is a variable of type } \sigma,\ \sigma \text{ any type} \} \text{ and } \to^\eta \triangleq \to^{\mathfrak{R}_\eta}$$

(In general, a subscript on a reduction set will be used as a superscript on $\to$ to indicate the corresponding top-level reduction.) The beauty of this approach to the definition of reductions is that the usual side condition for η-reduction, "$x \notin \mathrm{FV}(q)$", is not necessary because substitution renames bound variables to avoid the free variables of a term to which it is applied. Thus $\to^\eta$ is in fact the familiar top-level η-reduction. The reader may define sets $\mathfrak{R}_L$, $\mathfrak{R}_R$, and $\mathfrak{R}_P$ that generate the usual L-, R- and P-reductions.

How does β-reduction fit this scheme? The idea is to parameterize the reduction set by syntax as well as type. For example:

$$\mathfrak{R}_\beta \triangleq \{ \langle (\lambda x.M)q, [q/x]M \rangle : M \text{ is an expression, } q,\ x \text{ are variables} \}$$

The reader may verify that this corresponds to the definition of $\to^\beta$ given in the previous section. Aside from the unified approach, the reason for such a view of β-reduction will become more apparent when we consider more subtle versions of $\mathfrak{R}_\beta$.

The motivation for the terminology of this section is the following definition and result.

Definition A relation is *substitutive* $\overset{\text{def}}{\Leftrightarrow}$ $M \to N \Rightarrow sM \to sN$, for any substitution s.

□

Proposition 3.6 Given $\Re$, $\to^{\Re}$ is substitutive. If $\Re$ is substitutive, $\to^{\Re} = \Re$.

□

Finally, a *reduction* is a relation that is the congruence closure of a top-level reduction. Reductions are thus substitutive:

Proposition 3.7 If $\to^0$ is substitutive, so is $\to_0$.

□

3.4. Trivial Replacements

One of the conditions for Proposition 3.3 holds for all of the reductions in this paper.

Definition An expression is *trivial* $\overset{\text{def}}{\Leftrightarrow}$ it is a variable.

□

The reduction sets $\Re_\eta$, $\Re_L$, $\Re_R$, $\Re_P$ all have trivial replacements. This property enables a simple proof of one of the delayability conditions.

Proposition 3.8 Let $\Re_0$ have trivial replacements, and $\Re_1$ be arbitrary. Then $\to^0$ can be delayed past $\to_1$ via =.

□

4. Stratification

4.1. Positive β-reduction

In this subsection we shall consider a subrelation of β-reduction, as the first and most direct application of the ideas of the previous sections. We shall give all results without proofs, because they are special cases of more general results that we will be presenting in later subsections. The results given here may be proved without much difficulty; the only problem is in defining a relation for which *positive* weak commutativity holds. The key is this:

$$\Re_{\beta+} \triangleq \{ \langle (\lambda x.M)q, [q/x]M \rangle \in \Re_\beta : x \in \mathrm{FV}(M) \}$$

The remainder of $\to_\beta$ arises by union with $\to_{\beta 0}$, where:

$$\Re_{\beta 0} \triangleq \{ \langle (\lambda x.p)q, p \rangle : p,\ q, \text{ and } x \text{ are variables} \}$$

In later subsections, we shall consider the union with $\to_{\beta 0}$. It is useful to *stratify* $\Re_{\beta+}$ as follows:

$$\Re_{\beta+,\sigma} \triangleq \{ \langle (\lambda x.M)q, [q/x]M \rangle \in \Re_{\beta+} : \mathrm{type}(\lambda x.M) = \sigma \}$$

Our strategy is to extend a partial ordering on types to a well-ordering. Specifically:

Definition Inductively on type, $\sigma \sqsubseteq \tau \overset{\text{def}}{\Leftrightarrow}$

$$\sigma = \tau \text{ or } \tau = \tau_1 \to \tau_2 \text{ and } \sigma \sqsubseteq \tau_i,\ i = 1 \text{ or } 2.$$

□

The well-ordering on types becomes the index set I of Lemma 3. The condition $i<j$ in the hypothesis there will translate into $\sigma \not\sqsupseteq \tau$ here, so that any linearization of the partial ordering will suffice.

In the following summary of results, we shall use the convention $\to_\sigma \triangleq \to_{\beta+,\sigma}$ and $\to^\sigma \triangleq \to^{\beta+,\sigma}$.

The family $\{ \to_\tau \}_\tau$ has finite application.

For any type τ, $\to_\tau$ is well-founded and has finite degree.

If $\sigma = \sigma_1 \to \sigma_2$ and $\sigma_1, \sigma_2 \neq \tau$, then $\to_\sigma$ is delayable past $\to_\tau$ via $\to_\tau$.

If $\sigma \neq \tau$, then $\to_\sigma, \to_\tau$ is positive weak commutative.

For any type τ, $\to_\tau, \to_\tau$ is weak commutative.

Since $\sigma_1 \to \sigma_2 = \sigma \not\geq \tau \Rightarrow \sigma_1, \sigma_2, \sigma \neq \tau$, we can indeed apply Lemma 3, and conclude that $\to_{\beta+}$ is well-founded and Church-Rosser. The results listed above from an outline for the rest of this section.

4.2. Conjugation

Working through the proof of positive weak commutativity of $\to_\sigma, \to_\tau$ for $\sigma \neq \tau$ reveals that positivity results from the fact that no free variables are dropped by a reduction. This is not true for $\mathfrak{R}_{\beta 0}$, $\mathfrak{R}_L$, and $\mathfrak{R}_R$, and positivity fails in these cases, though we can prove weak commutativity. The trick is to augment these reductions by others, in order to get delayability and positive weak commutativity. As an example, we define $\mathfrak{R}_\gamma$ so that when $y \notin FV(M) \cup FV(Q)$ and $x \in FV(M)$:

$$(\lambda y \lambda x.M)NQ \to^\gamma (\lambda y.[Q/x]M)N$$

In other words, we allow the positive β-reduction without doing the $\beta 0$-reduction to expose the abstraction $\lambda x.M$. Note that this rule is sound, because it is given by

$$(\lambda y \lambda x.M)NQ \to_{\beta 0} (\lambda x.M)Q \to^{\beta 0} [Q/x]M \, (\to_{\beta 0})^{-1} \, (\lambda y.[Q/x]M)N,$$

where $^{-1}$ denotes the inverse relation. The sequence $\to_{\beta 0} \to^{\beta +} (\to_{\beta 0})^{-1}$ yields the term conjugation.

The reasoning described above, and the theory that we will develop in this section, apply much more generally than to just $\mathfrak{R}_{\beta 0}$. We will work with a reduction set $\mathfrak{R}_0$ that satisfies certain properties that we will introduce as the section progresses; $\mathfrak{R}_{\beta 0}$ will have the desired properties. The first of these is that $\mathfrak{R}_0$ must have trivial replacements, which we previously saw was true of $\mathfrak{R}_{\beta 0}$.

To understand the next property, review the above expansion of $\to^\gamma$. In the forward direction of $\to^{\beta 0}$, the type of the replacement was type($\lambda x.M$); in the reverse direction, the type of the replacement was type($[Q/x]M$)$\subset$type($\lambda x.M$). This causes no difficulty, because $\mathfrak{R}_{\beta 0}$ is parameterized freely over all types. The following definitions capture the essence of this observation.

Definition Given $\mathfrak{R}_0$, its *strata* are given by:

$$\mathfrak{R}_{0,\sigma} \triangleq \{ \langle P, Q \rangle \in \mathfrak{R}_0 : \text{type}(P) = \sigma \}$$

□

Definition A reduction set $\mathfrak{R}_0$ with trivial replacements is *uniform* (over types) $\overset{\text{def}}{\Leftrightarrow}$ for types σ, τ, these is a one-one correspondence between $\mathfrak{R}_{0,\sigma}$ and $\mathfrak{R}_{0,\tau}$, $\langle P_\sigma, p_\sigma \rangle \leftrightarrow \langle P_\tau, p_\tau \rangle$ such that P_σ and P_τ have the same syntactic shape and the same types for free variables other than p_σ and p_τ.

□

In order to make this notion completely rigorous, we would have to set up correspondences between the constants that appear in reductions. For example, we clearly mean that the correspondence in $\mathfrak{R}_L$ be left (pair p_σ y) $\leftrightarrow$ left (pair p_τ y). The two appearances of "left" have different type, but there is only one "left" for each pair of types σ, type(y). In $\mathfrak{R}_{\beta 0}$, all that is necessary is to change the type of "p": $(\lambda x.p_\sigma)q \leftrightarrow (\lambda x.p_\tau)q$. Since well-formedness on types requires only that type(x)=type(q), both these are legitimate typed expressions.

Throughout this section, the types will usually not be affixed to a symbol. Rather, the correspondences will be indicated by $\langle P, p \rangle \leftrightarrow \langle \hat{P}, \hat{p} \rangle$, and the types will be clear from context. In fact, if type(p) = $\sigma_1 \to \sigma_2$, we will always have type($\hat{p}$) = σ_2.

Proposition 4.1 Suppose $\mathfrak{R}_0$ has trivial replacements and is uniform. Given $P \twoheadrightarrow_0 p$, and a variable $\hat{p}$. Then $\exists \hat{P}$ with the same syntactic shape as P, where $\hat{P} \twoheadrightarrow_0 \hat{p}$.

Proof We work by induction on the length of $P \twoheadrightarrow_0 p$. If this has length 0, then $P = p$, and we let $\hat{P} \triangleq \hat{p}$. Otherwise, $P \to_0 Q \twoheadrightarrow_0 p$, and by induction there is $\hat{Q} \twoheadrightarrow_0 \hat{p}$. The reduction $P \to_0 Q$ takes place at top level at some subexpression of P; this subexpression may be written in the form sP_0, where $\langle P_0, p_0\rangle \in \Re_0$, and Q has a subexpression of the form sp_0. Compare $\hat{Q}$ with Q. This subexpression will have the same shape, though perhaps different type, and may thus be written $\hat{s}\hat{p}_0$, where $\hat{p}_0$ has the appropriate type, $\langle \hat{P}_0, \hat{p}_0\rangle \in \Re_0$ (by uniformity), and $\hat{s}$ has the same shape as s. Form $\hat{P}$ from $\hat{Q}$ by replacing the subexpression $\hat{s}\hat{p}_0$ with $\hat{s}\hat{P}_0$; this has the same shape as P.

□

The $\hat{P}$ given by this Proposition may be used whenever we have $P \twoheadrightarrow_0 p$; we do not refer to this result explicitly.

Definition Given a uniform reduction set $\Re_0$ with trivial replacements, its *conjugate* is $\Re_\gamma \triangleq \cup_\sigma \Re_{\gamma,\sigma}$ where

$$\Re_{\gamma,\sigma} \triangleq \{ \langle ([\lambda x.M/p]P)q, [[q/x]M/\hat{p}]\hat{P}\rangle : P \twoheadrightarrow_0 p, x \in \mathrm{FV}(M), \text{ and type}(p) = \sigma \}$$

□

Note that $\Re_{\beta+} \subseteq \Re_\gamma$ for any $\Re_0$, because $p \twoheadrightarrow_0 p$. The general soundness argument is that:

$$([\lambda x.M/p]P)q \twoheadrightarrow_0 (\lambda x.M)q \rightarrow^{\beta+} [q/x]M \; (\twoheadrightarrow_0)^{-1} \; [[q/x]M/\hat{p}]\hat{P}$$

With suitable conditions on $\Re_0$, the results for $\Re_{\beta+}$ can be generalized to $\Re_\gamma$, as we shall see.

For the remainder of this section, we shall let $\rightarrow^\tau$ be the top-level reduction induced by the stratum $\Re_{\gamma,\tau}$, and $\rightarrow_\tau$ its congruence closure. This gives rise to the family of relations $\{ \rightarrow_\tau \}_\tau$ where $\rightarrow_\infty$ for the family is $\rightarrow_\gamma$. We begin supplying the hypotheses for Lemma 3 with the following two results.

Proposition 4.2 For any τ, $\{\rightarrow_\tau\}_\tau$ has finite application.

Proof Given M, $\rightarrow_\gamma$ does not produce any new types. Thus, any $\rightarrow_\tau$ steps in a derivation of M will have τ among the types of subexpressions of M, and this set is finite.

□

Proposition 4.3 For any τ, $\rightarrow_\tau$ has finite degree.

Proof By Proposition 3.5, we need prove only that $\rightarrow^\tau$ has finite degree. If $M \rightarrow^\tau M'$, then M is an application M_1M_2, and there are only a finite number of ways to write $M_1 = s([\lambda x.M/p]P)$ for varying s, M, and P, not counting choices of variable names for p and elements of Dom(s).

□

4.3. Linear Expressions

This section introduces one of the restrictions that we shall make on $\Re_0$, and provides two basic results.

Definition An expression P is *linear wrt* a variable $x \overset{\text{def}}{\Leftrightarrow} x$ occurs once in P; P is *linear wrt* a set of variables $\overset{\text{def}}{\Leftrightarrow} P$ is a linear wrt x, $\forall x \in S$; P is *linear wrt* a substitution $s \overset{\text{def}}{\Leftrightarrow} P$ is linear wrt Dom(s). If P is linear wrt FV(P), we say simply that P is linear.

□

We shall be interested in reduction sets with linear patterns. The reader may check that $\Re_L$, $\Re_R$, $\Re_\pi$ and $\Re_{\beta 0}$ all have linear patterns, and that $\Re_P$ does not. As we defined $\Re_\beta$ and $\Re_\gamma$, they do not have linear patterns. We can change this by fiat.

Definition Let $\Re^{\text{old}}_{\gamma,\sigma}$ and $\Re^{\text{old}}_\gamma$ be the previous definitions; redefine $\Re_\gamma \triangleq \bigcup_\sigma \Re_{\gamma,\sigma}$ where

$$\Re_{\gamma,\sigma} \triangleq \{ \langle P, Q\rangle \in \Re^{\text{old}}_{\gamma,\sigma} : P \text{ is linear} \}$$

□

For the $\Re_0$ in which we are interested, this redefinition makes no difference.

Proposition 4.4 If $\Re_0$ has linear patterns, the old and new definitions of $\Re_\gamma$ yield the same $\rightarrow^\gamma$.

○

The importance of the linear property is its connection with substitution. The following is the first of several results concerning reductions into or out of an expression of the form sM.

Lemma 4 Let $\Re_0$ have linear patterns and trivial replacements, and let $N \rightarrow_0 sM$ where $\mathrm{Dom}(s) \cap \mathrm{FV}(N) = \phi$ and M is linear wrt s. Then either:

$N = s'M$ where $s' \rightarrow_0 s$, or

$N = sM'$ where $M' \rightarrow_0 M$ and M' is linear wrt s.

○

For the second result, there are some preliminaries.

Definition Given substitutions s and t, $s \circ t \triangleq [s(tx)/x]_{x \in \mathrm{Dom}(t)}$.

□

Proposition 4.5 For an expression M with $\mathrm{Dom}(t) = \mathrm{FV}(M)$: $(s \circ t) = (s(t(M))$.

□

Definition P_0 is a *factor* of $P \overset{\mathrm{def}}{\Leftrightarrow} \exists\, P_1, p : P = [P_0/p]P_1$. It is called *proper* $\overset{\mathrm{def}}{\Leftrightarrow} P_0 \neq P$.

□

Definition An expression M is *open* $\overset{\mathrm{def}}{\Leftrightarrow} \mathrm{FV}(M) \neq \phi$; otherwise, it is *closed*. A substitution s is *open* (resp., *closed*) $\overset{\mathrm{def}}{\Leftrightarrow}$ each element of Range(s) is open (resp., closed).

□

Definition For an expression P:

$T(P) \triangleq \{\ \mathrm{type}(P_0) : P_0$ is a non-trivial open proper factor of $P\ \}$.

For a substitution s:

$T(s) \triangleq \{\ \mathrm{type}(x) : x \in \mathrm{Dom}(s)\ \}$

□

Lemma 5 Let $sM = tP$ where s is open, M is non-trivial and P is linear, $\mathrm{Dom}(t) = \mathrm{FV}(P)$, and $T(s) \cap T(P) = \phi$. Then $\exists u : M = uP$ and $t = s \circ u$.

○

4.4. Delayability of $\rightarrow_0$

We now have enough machinery to relate $\rightarrow_0$, $\rightarrow_\beta$, and $\rightarrow_\gamma$. Suppose we can prove that $\rightarrow_\gamma$ is well-founded. In this subsection, we show how to conclude that $\rightarrow_0 \cup \rightarrow_\gamma$ is well-founded. Since $\rightarrow_\beta \subseteq \rightarrow_\gamma$, we will also have $\rightarrow_0 \cup \rightarrow_\beta$ well-founded.

Proposition 4.6 Let $\Re_0$ have linear patterns and trivial replacements. Then $\rightarrow_0$ is delayable past $\rightarrow_\sigma$ via =.

○

Proposition 4.7 Let $\rightarrow_0$ have linear and non-trivial patterns and trivial replacements. Assume that $\rightarrow_\gamma$ is well-founded. Then $\rightarrow_0 \cup \rightarrow_\gamma$ is well-founded.

Proof By non-trivial patterns and trivial replacements, $\rightarrow_0$ is well-founded. then use the previous result, the hypothesis for $\rightarrow_\gamma$, and Corollary 1.

□

In the remainder of the section, then, our sole concern is the well-foundedness of $\rightarrow_\gamma$.

4.5. Prime Expressions

Lemma 5 is very useful, but only indirectly. We need a way to characterize $T(P)$ and $T(s)$ so that we can prove null intersection. The following idea is sufficient for many purposes.

Definition An expression P is *prime* $\overset{\text{def}}{\Leftrightarrow}$ for every non-trivial open proper factor N of P: type(N) $\supset$ type(P).

□

(Note that P is prime $\Leftrightarrow$ ($\sigma \in T(P) \Rightarrow \sigma \supset$type($P$)).) The relations $\Re_{\beta 0}$, $\Re_{\eta}$, $\Re_{L}$, and $\Re_{R}$ are all prime; $\Re_{P}$ is not. As with the linearity, $\Re_{\beta}$ and $\Re_{\gamma}$ are not prime, but we may redefine them to be so. We do this after giving the basic result:

Proposition 4.8 Let $sM = tP$, where s is open, M is non-trivial and P is linear and prime. Suppose $\sigma \in T(s)$ has $\sigma \not\supset$type(P). Then $\exists u$: $M = uP$ and $t = s \circ u$.

Proof To use Lemma 5, we must verify only that $T(s) \cap T(P) = \phi$:

$$\sigma \in T(P) \text{ and } \sigma \in T(s) \Rightarrow \sigma \supset \text{type}(P) \text{ and } \sigma \not\supset \text{type}(P), \text{ contradiction.}$$

The reasons are that P is prime, and the hypothesis.

□

We once again redefine $\Re_{\gamma}$ and its strata.

Definition Given $\Re_0$ with trivial replacements, let $\Re^{\text{old}}_{\gamma,\sigma}$ be the previous definition of $\Re_{\gamma,\sigma}$, and redefine $\Re_{\gamma} \triangleq \cup_{\sigma} \Re_{\gamma,\sigma}$, where $\Re_{\gamma,\sigma} \triangleq \{ \langle P, Q \rangle \in \Re^{\text{old}}_{\gamma,\sigma} : P \text{ is prime} \}$ □

Proposition 4.9 Let $\Re_0$ have non-trivial, linear, and prime patterns, and trivial replacements. Then the old and new definitions of $\Re_{\gamma}$ give rise to the same $\to^{\gamma}$.

○

Throughout the remainder of this section, we will let $\to^{\sigma} \triangleq \to^{\gamma,\sigma}$ and $\to_{\sigma} \triangleq \to_{\gamma}{}^{\sigma}$. We close this subsection with a key result and one of its consequences.

Lemma 6 Let $sM \to_{\tau} N$ where M is linear wrt s and $\sigma \in T(s) \Rightarrow \sigma \not\geq \tau$. Then either:

$N = sM'$ where $M \to_{\tau} M'$, or

$N = s'M$ where $s \to_{\tau} s'$.

○

Corollary 4 Let $sM \to_{\tau} N$ where $\sigma \in T(s) \Rightarrow \sigma \not\geq \tau$. Then either:

$N = sM'$ where $M \to_{\tau} M'$, or

$N \twoheadrightarrow_{\tau} s'M$ where $s \to_{\tau} s'$

○

4.6. Well-foundedness

The main result here not only supplies one of the hypotheses for Lemma 3, it also is necessary in some of the commutativity and delayability results for strata of $\Re_{\gamma}$. In this subsection, the term *normalizable* will mean not having an infinite $\to_{\tau}$ derivation and will apply to both expressions and substitutions. We begin with an auxiliary result.

Proposition 4.10 Given τ, s, and M, suppose that $\sigma \in T(s) \Rightarrow \sigma \not\geq \tau$, and that s and M are normalizable. Then sM is normalizable.

○

This provides a simple proof of the desired result.

Proposition 4.11 Let $\mathfrak{R}_0$ be uniform with linear and prime patterns. For any type τ, $\to_\tau$ is well-founded.

Proof By a simple induction on terms argument, we may reduce the proof to showing the normalizability of an application $N = N_1N_2$, where N_1 and N_2 are normalizable, and at least one step in the derivation is at top level, i.e.:

$$N_1 \twoheadrightarrow_\tau [\lambda x.M/p]P \text{ where } x \in \mathrm{FV}(M),\ P \twoheadrightarrow_0 p, \text{ and type}(p) = \sigma$$

$$N_2 \twoheadrightarrow_\tau Q$$

$$N_1N_2 \twoheadrightarrow_\tau ([\lambda x.M/p]P)Q \to^\tau [[Q/x]M/\hat{p}]\hat{P}$$

Since N_1 and N_2 are normalizable, so are Q, M, and P. Since $T([Q/x]) = \{\text{type}(x)\}$ and $\text{type}(x) \subset \tau$, the previous result says that $[Q/x]M$ is normalizable, and thus, so is $[[Q/x]M/\hat{p}]$. But $T([[Q/x]M/\hat{p}]) = \{\text{type}(\hat{p})\}$ and $\text{type}(\hat{p}) \subset \tau$. Further $\hat{P}$ has the same derivations as P, and is thus normalizable. Again using the previous result, $[[Q/x]M/\hat{p}]\hat{P}$ is normalizable. Thus all derivations from N lead to a normalizable expression, and N is normalizable.

□

As a direct consequence of this result and Proposition 4.6, Lemma 3 yields:

Proposition 4.12 For the same $\mathfrak{R}_0$ as above, and any type τ, $\to_\tau \cup \to_0$ is well-founded.

○

4.7. Irreducible Expressions

It remains to prove positive weak commutativity and delayability for the pairs $\to^\sigma, \to_\tau$ and $\to_\sigma, \to^\tau$. For the pair $\to^\sigma, \to_\tau$, primality is the key for both conditions. For the pair $\to_\sigma, \to^\tau$, we need a slightly different condition. With apologies for the nomenclature:

Definition An expression P is *irreducible* $\overset{\text{def}}{\Leftrightarrow}$ P has no bound variables, and every non-trivial subexpression P_0 of P has $\text{type}(P_0) \supseteq \text{type}(P)$.

□

Neither primality nor irreducibility implies the other; primality considers fewer subexpressions, but requires strict ordering on types. The reduction sets $\mathfrak{R}_{\beta 0}$, $\mathfrak{R}_L$ and $\mathfrak{R}_R$ have irreducible patterns. Those of $\mathfrak{R}_\eta$ are not irreducible, because $\text{type}(px) \not\supseteq \text{type}(\lambda x.px)$; we shall see later that η-reduction can be handled without stratification and conjugation.

Our goal is to show that we may once again modify the definition of $\mathfrak{R}_{\gamma,\sigma}$, this time adding the additional constraint that "P" be irreducible.

Proposition 4.13 Let $\mathfrak{R}_0$ have linear and irreducible patterns and trivial replacements. If $P \twoheadrightarrow_0 p$, then $\exists s, P_0, p_0$:

$$P = sP_0,\ P_0 \twoheadrightarrow_0 p_0, \text{ and } sp_0 = p.$$

$$P_0 \text{ is irreducible and linear wrt } s.$$

○

For the last time:

Definition Given $\mathfrak{R}_0$ with trivial replacements, let $\mathfrak{R}^{\text{old}}_{\gamma,\sigma}$ be the previous definition of $\mathfrak{R}_{\gamma,\sigma}$, and redefine $\mathfrak{R}_\gamma \triangleq \cup_\sigma \mathfrak{R}_{\gamma,\sigma}$, where:

$$\mathfrak{R}_{\gamma,\sigma} \triangleq \{ \langle ([\lambda x.M/p]P)q,\ [[q/x]M/\hat{p}]\hat{P} \rangle \in \mathfrak{R}^{\text{old}}_{\gamma,\sigma} : P \text{ is irreducible} \}$$

□

Proposition 4.14 The new and old definitions of $\Re_\gamma$ yield the same $\rightarrow^\gamma$.

○

4.8. Commutativity

We first examine the simplest of problems—how might we prove that $\rightarrow^0, \rightarrow^0$ is weak commutative? For the reductions we have been considering, this follows immediately from:

Proposition 4.15 Suppose that $\rightarrow^0$ is a function on its domain. Then $\rightarrow^0, \rightarrow^0$ is weak commutative.

□

Next, we consider ways of proving that $\rightarrow_0$ is weak commutative, for example, how might we prove this for $\rightarrow_{\beta 0}, \rightarrow_{\beta 0}$? Looking at the rule $(\lambda x.p)q \rightarrow_{\beta 0} p$, it is evident that no matter what is substituted for p or q, any non top-level $\beta 0$-reduction from the expression will occur within a substituted expression—$\lambda x.P \not\rightarrow_0$ anything, for any P. This observation motivates:

Definition A set of expressions P is *independent* $\overset{\text{def}}{\Leftrightarrow}$ $\forall P,Q \in \Re$ and substitutions s and t with $\mathrm{Dom}(s) = \mathrm{FV}(P)$, if M is a non-trivial proper subexpression of P, then $sM \neq tQ$.

□

It is easily checked that $\Re_{\beta 0} \cup \Re_L \cup \Re_R$ has independent patterns; if either $\Re_P$ or $\Re_\eta$ are adjoined to this set, independence is lost.

Proposition 4.16 If $\Re_0$ has linear independent patterns and if $\rightarrow^0, \rightarrow^0$ is weak commutative, then $\rightarrow_0, \rightarrow_0$ is weak commutative.

○

In our applications, $\rightarrow_0$ will be bounded, and Corollary 2 says that $\twoheadrightarrow_0, \twoheadrightarrow_0$ is commutative.

We next study the commutativity of $\twoheadrightarrow_\beta, \twoheadrightarrow_0$. This not only leads to the commutativity of $\twoheadrightarrow_{\beta +} \cup \twoheadrightarrow_0$, it also figures in the proof of delayability in the next subsection. In order to get the result, we introduce another condition.

Definition An expression is *anchored* $\overset{\text{def}}{\Leftrightarrow}$ no free variable appears as an operator (the first subexpression of an application).

□

Proposition 4.17 Given $\Re_0 \supseteq \Re_{\beta 0}$ with linear, irreducible, independent and anchored patterns and trivial replacements, and $\rightarrow_0, \rightarrow_0$ weak commutative. For any τ, $\rightarrow_\tau \cup \rightarrow_0, \rightarrow_0$ is weak commutative.

○

Proposition 4.18 Under the same conditions as above, and for any type τ: $(\rightarrow_\tau \cup \rightarrow_0)^*$, $\twoheadrightarrow_0$ is commutative.

Proof Use Proposition 4.12 and Corollary 2.

□

Finally, we consider the commutativity relations within $\Re_\gamma$.

Proposition 4.19 Given a uniform reduction set $\Re_0$ with linear and prime patterns. Assume that $\twoheadrightarrow_0, \twoheadrightarrow_0$ is commutative. For types σ, τ with $\sigma \not\geq \tau$, $\rightarrow^\sigma, \rightarrow_\tau$ is positive weak commutative.

○

Proposition 4.20 Given a reduction set $\Re_0$ with irreducible linear patterns and trivial replacements. If $\sigma \not\geq \tau$, then $\rightarrow_\sigma, \rightarrow^\tau$ is positive weak commutative.

○

4.9. Delayability

In this section we consider the delayability of $\to_\sigma$ past $\to_\tau$.

Proposition 4.21 Let $\Re_0$ have linear and prime patterns and trivial replacements. Given types σ, τ with $\sigma \not\geq \tau$, $\to^\sigma$ is delayable past $\to_\tau$ via $\to_\tau$.

O

Proposition 4.22 Let $\Re_0$ have linear and irreducible patterns and trivial replacements. Given types σ,τ with $\sigma \not\geq \tau$, $\to_\sigma$ is delayable past $\to^\tau$ via =.

O

4.10. Summary

We now have all the pieces for applying Lemma 3.b. We state the following only to summarize the results.

Theorem 1 Given $\Re_0$, assume:

> The patterns of $\Re_0$ are linear, prime, irreducible, anchored, and form an independent set.
>
> $\Re_0$ has trivial replacements.
>
> $\Re_{\beta 0} \subseteq \Re_0$.

Then $\to_\beta \cup \to_0$ is well-founded and its reflexive transitive closure is Church-Rosser.

□

All of the conditions on the patterns of $\Re_0$, except for independence, are easily checked by looking at individual rules; independence is generally not difficult to verify. Since $\Re_{\beta 0}$ meets all the requirements, this result gives a new proof of the well-foundedness and Church-Rosser properties of $\to_\beta$.

5. Delayability past β-reduction

In this section we consider reductions $\to_0$ that are delayable past β-reduction, and more generally, past reductions $\to_1$ that can be proved well-founded by the methods of the previous section. Examples of $\to_0$ are $\to_\eta$ and $\to_P$. In each case, the delayability of $\to^0$ past $\to_1$ via = follows from Proposition 3.8. By Proposition 3.3, all that we need consider in this section is the delayability of $\to_{\bar{0}}$ past $\to^1$ via =.

5.1. η-reduction

The delayability of $\to_\eta$ past $\to_\beta$ via = is essentially classical; it is called "postponement" in the literature, for example, see [1], p. 382. Let $\to_1 = \to_\beta \cup \to_2$ where $\to_2$ arises from $\Re_2$. We seek a simple condition on the patterns of $\Re_2$ that guarantees the delayability of $\to_\eta$ past $\to_1$.

Definition An expresssion P is *η-delayable* $\overset{\text{def}}{\Leftrightarrow}$ every non-trivial subexpression in operand position has a non-arrow type.

□

Proposition 5.1 Let $\Re_2$ have linear and η-delayable patterns, and $\to_1 \triangleq \to_\beta \cup \to_2$. Then $\to_\eta^-$ is delayable past $\to_\beta \cup \to^2$ via =.

O

In our example, $\to_2 = \to_L \cup \to_R$; the only non-trivial subexpression in operand position of a pattern of $\Re_L \cup \Re_R$ is of the form "pair x y", and has type $\sigma\times\tau$, not an arrow type. Hence, we conclude that $\to_\beta \cup \to_L \cup \to_R \cup \to_\eta$ is well-founded.

5.2. λ-free Replacements

The delayability of $\to_P$ past $\to_\beta$ follows from the fact that $\Re_P$ has replacements with the following property.

Definition An expression Q is *λ-free* $\overset{def}{\Leftrightarrow}$ for every substitution s, sQ is not an abstraction.

□

Proposition 5.2 Let $\Re_0$ have λ-free replacements. Then $\to_{\bar{0}}$ is delayable past $\to^\beta$.

○

This result shows that $\to_P$ can be delayed past $\to_\beta$—the "Q" in a P-reduction has type $\sigma\times\tau$, not the type of an abstraction. Since $\to_\eta$ is also delayable past $\to_\beta$, we have $\to_\eta\cup\to_P$ delayable past $\to_\beta$. The reduction $\to_\eta\cup\to_P$ reduces the size of the expression, and so it is well-founded. This means that $\to_\beta\cup\to_\eta\cup\to_P$ is well-founded. The proposition of this subsection is also useful when $\Re$ has replacements that are applications (which are λ-free regardless of type), but to get delayability of $\to_0$ past $\to_\beta$, we can no longer rely on Proposition 3.8.

5.3. Pairing Interaction

We still have not shown that $\to_\beta\cup\to_\eta$, together with all three pairing reductions, is well-founded. To show this all we must prove is:

Proposition 5.3 $\to_{\bar{P}}$ is delayable past $\to^L\cup\to^R$ via =.

Proof Consider $N \to_{\bar{P}}$ left (pair M_1 M_2) $\to^L M_1$. The only interesting case is that $\to_{\bar{P}}$ might supply "pair M_1 M_2". But then $\to^L$ leads to the same expression as $\to_{\bar{P}}$:

$$N = \text{left (pair (left (pair } M_1\ M_2)) \text{ (right (pair } M_1\ M_2))) \to^L \text{left (pair } M_1\ M_2) \to^L M_1$$

The $\to_P$ reduction simply disappears, and $\to_{\bar{P}}$ is delayable past $\to^L$. Symmetrically, we can prove that it is delayable past $\to^R$, and by Proposition 2.1, $\to_P$ is delayable past $\to^L\cup\to^R$ via =.

□

Thus $\to_P$ is delayable past $\to_\beta\cup\to_L\cup\to_R$ via =. Since $\to_\eta$ is also, so is $\to_P\cup\to_\eta$, and this relation is well-founded. This finally proves the well-foundedness of $\to_\beta\cup\to_L\cup\to_R\cup\to_P\cup\to_\eta$.

Bibliography

1. Barendregt, H.P. *The Lambda Calculus.* North Holland, 1981.

2. Bercovici, I. Tait's method and strong normalizability of pairing rules. Unpublished

3. Gandy, R.O. Proofs of Strong Normalization. In J.P. Seldin and J.R. Hindley, editors, *To H.B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*, pages 457-477. Academic Press, 1980.

4. Karr, M. Tait's method and surjective pairing. Unpublished

5. Karr, M. "Delayability" in Proofs of Strong Normalizability in the Typed Lambda Calculus. Unpublished

6. Klop, J.W. *Combinatory Reduction Systems.* Mathematisch Centrum Amsterdam, 1980.

7. Nederpelt, R.P. *Strong Normalization in a Typed λ-Calculus with Lambda Structured Types.* Ph.D. Th., The University of Technology, Eindhoven, 1973.

8. Tait, W.W. Intensional Interpretations of Functionals of Finite Type I. *J. of Symbolic Logic 32* (1967), 198-212.