A reduction relation for which postponement of K-contractions, Conservation and Preservation of Strong Normalisation hold*

Fairouz Kamareddine†

March 13, 1996

Abstract

Postponement of β_K-contractions and the conservation theorem do not hold for ordinary β but have been established by de Groote for a mixture of β with another reduction relation. In this paper, de Groote's results are generalised for a single reduction relation β_e which generalises β. This then is used to solve an open problem of β_e: the Preservation of Strong Normalisation[1].

Keywords: Generalised β-reduction, Postponement of K-contractions, Generalised Conservation, Preservation of Strong Normalisation.

1 Introduction

1.1 Background and Motivation

In the term $((\lambda_x.\lambda_y.N)P)Q$, the function starting with λ_x and the argument P result in the redex $(\lambda_x.\lambda_y.N)P$. It is also the case that the function starting with λ_y and the argument Q will result in another redex when the first redex is contracted. This idea has been exploited by many researchers and reduction has been extended so that the generalised redex based on the matching λ_y and Q is given the same priority as the other redex. Reasons for generalising redexes and β-reduction are numerous and have ranged between theoretical and practical. Here are a few attempts at generalising reduction and at the reasons of such an extension:

$$(\theta) \qquad ((\lambda_x.N)P)Q \to (\lambda_x.NQ)P$$

$$(\gamma) \qquad (\lambda_x.\lambda_y.N)P \to \lambda_y.(\lambda_x.N)P$$

$$(\gamma_C) \qquad ((\lambda_x.\lambda_y.N)P)Q \to (\lambda_y.(\lambda_x.N)P)Q$$

*Preservation of Strong Normalisation for generalised reduction proved to be a difficult problem to establish. Joe Wells told me of the result of de Groote on conservation and that it might help me solve the problem. Without Joe's observation, PSN would probably have remained a difficult problem to solve. I am grateful for him for the discussions we had and for his patience when I needed help on latex. I am also grateful for Alejandro Rios for his comments on the paper. This work is partially supported by EPSRC grant number GR/K25014 and was carried out at Boston University to whom and especially to Assaf Kfoury, I am grateful.

†Department of Computing Science, 17 Lilybank Gardens, University of Glasgow, Glasgow G12 8QQ, Scotland, *email*: fairouz@dcs.glasgow.ac.uk, fax + 44 41 -3304913.

[1] I was told by Rob Nederpelt that in [Nederpelt 73], he assumed that the preservation of Strong Normalisation for β_e is easy to establish yet when editing [NGV 94], he retracted his assumption.

All these (related) rules attempt to make more redexes visible. γ_C for example, makes sure that λ_y and Q form a redex even before the redex based on λ_x and P is contracted. Due to compatibility, γ implies γ_C. Moreover, $((\lambda_x.\lambda_y.N)P)Q \to_\theta (\lambda_x.(\lambda_y.N)Q)P$ and hence both θ and γ_C put λ adjacently next to its matching argument. One can say that θ moves the argument next to its matching λ whereas γ_C moves the λ next to its matching argument. Hence, θ can be equally applied to explicitly and implicitly typed systems. The transfer of γ or γ_C to explicitly typed systems is not straightforward however, since in explicitly typed systems, the type of y may be affected by the reducible pair λ_x, P. For example, it is fine to write $((\lambda_{x:*}.\lambda_{y:x}.y)z)u \to_\theta (\lambda_{x:*}.(\lambda_{y:x}.y)u)z$ but it is not fine to write $((\lambda_{x:*}.\lambda_{y:x}.y)z)u \to_{\gamma_C} (\lambda_{y:x}.(\lambda_{x:*}.y)z)u$. For this reason, we study θ-like rules for generalised reduction in this paper. Now, we discuss where generalised reduction has been used ([KW 95b] provides a more detailed comparison).

[Reg 92] introduces the notion of a *premier redex* which is similar to the redex based on λ_y and Q above (and which is called *extended redex* in the paper). [Reg 94] uses θ and γ (and calls the combination σ) to show that the perpetual reduction strategy finds the longest reduction path when the term is Strongly Normalising (SN). [Vid 89] also introduces reductions similar to those of [Reg 94]. Furthermore, [KTU 94] uses θ (and other reductions) to show that typability in ML is equivalent to acyclic semi-unification. [SF 92] uses a reduction which has some common themes to θ. [Nederpelt 73] and [dG 93] use θ whereas [KW 95a] uses γ to reduce the problem of strong normalisation for β-reduction to the problem of weak normalisation for related reductions. [KW 94] uses amongst other things, θ and γ to reduce typability in the rank-2 restriction of system F to the problem of acyclic semi-unification. [AFM 95] uses θ (which they call "let-C") as a part of an analysis of how to implement sharing in a real language interpreter in a way that directly corresponds to a formal calculus. [KN 95] uses a more extended version of θ where Q and N are not only separated by the redex $(\lambda_x.N)P$ but by many redexes (ordinary and generalised).

There are other reasons for using generalised reduction than those mentioned above. [KN 95] showed that generalised reduction makes more redexes visible and hence allows for more flexibility in reducing a term. [BKN 9y] showed that with generalised reduction one may indeed avoid size explosion without the cost of a longer reduction path and that λ-calculus can be elegantly extended with definitions which result in shorter type derivation.

All the research mentioned above is a living proof for the importance and usefulness of generalised reduction (from now on, β_e). For this reason, properties of this reduction must be studied. Confluence of β_e is a direct consequence of the fact that $M =_\beta N \Leftrightarrow M =_{\beta_e} N$. Subject reduction for β_e has been established in [BKN 9y] (with the condition that *explicit definitions* must be added for some systems of the cube). Strong Normalisation of β_e has been established for the whole Cube (with or without definitions) in [BKN 9y]. One important property however, the *Preservation of Strong Normalisation* (PSN) of β_e has remained open. This property is: if M is strongly normalising for ordinary β-reduction (written M is β-SN), then M remains strongly normalising for generalised reduction β_e (i.e. M is also β_e-SN). PSN makes β_e a useful extension of β. This parallels the work on extending λ-calculi with explicit substitutions which satisfy the PSN property.

1.2 Contributions of this paper and related work

Let us recall the three basic reduction rules of the λ-calculus ($FV(M)$ stands for the free variables of M):

(β)	$(\lambda_x.M)N \to M[x := N]$	
(β_I)	$(\lambda_x.M)N \to M[x := N]$	if $x \in FV(M)$
(β_K)	$(\lambda_x.M)N \to M$	if $x \notin FV(M)$

Redexes based on the β_I rule are called β_I- or I-redexes. Similarly, those based on the β_K rule are called β_K- or K-redexes. For any relation r, we write r_K and r_I for the corresponding K- and I-reductions.

In this paper, we show that the generalised reduction β_e satisfies PSN. We do this by showing the *postponement of K-contractions* and *conservation* for β_e. These two latter properties are important on their own since the first says that when reducing a term to a normal form, we can first reduce all the I-redexes and then we can reduce all the K-redexes; the second says that if a term is I-normalising, then it is strongly normalising. Of course both properties fail for ordinary β. For example, $(\lambda_y.(\lambda_x.x))MN \to_{\beta_K} (\lambda_x.x)N \to_{\beta_I} N$ and it is impossible to β_I-reduce $(\lambda_y.(\lambda_x.x))MN$. Moreover, $((\lambda_x.\lambda_y.y(\lambda_z.zz))u)\lambda_z.zz$ is β_I-normalising but not strongly β-normalising.

Attempts have been made at establishing some reduction relations for which postponement of K-contractions and conservation hold ([Bar 84] and [dG 93]). The picture is as follows (-N stands for normalising and $r \in \{\beta_I, \theta_K\}$):

(β_K-postponement for r)	If $M \to_{\beta_K} N \to_r O$ then $\exists P$ such that $M \twoheadrightarrow^+_{\beta_I\theta_K} P \twoheadrightarrow_{\beta_K} O$
(Conservation for β_I)	If M is β_I-N then M is β_I-SN
(Conservation for $\beta + \theta$)	If M is $\beta_I\theta_K$-N then M is β-SN

Conservation for β_I is found in [Bar 84]. Conservation for $\beta + \theta$ and β_K-postponement for $r \in \{\beta_I, \theta_K\}$ are established in [dG 93]. However, de Groote does not produce these results for a single reduction relation, but for β in which another relation (θ) is used. This paper establishes β_K-postponement and conservation for a single relation β_e and is hence the first to do so. These properties for β_e are as follows:

(β_{eK}-postponement for β_e)	If $M \to_{\beta_{eK}} N \to_{\beta_{eI}} O$ then $\exists P$ such that $M \to_{\beta_{eI}} P \twoheadrightarrow^+_{\beta_{eK}} O$
(Conservation for β_e)	If M is β_{eI}-N then M is β_e-SN

To show PSN, we show that if $M \twoheadrightarrow_F N$ (using the perpetual strategy) and if N is β_{eI}-N then M is β_{eI}-N. Now, we take M which is β-SN, and its perpetual path to its normal form N. As N is β_{eI}-N, then M is β_{eI}-N and hence by conservation, M is β_e-SN.

Both our postponement and generalised conservation are important because here we have the first reduction relation which generalises β (yet $M =_\beta N \Leftrightarrow M =_{\beta_e} N$) and which satisfies them. The most important result of this paper however, is PSN: M is β-SN $\Leftrightarrow$ M is β_e-SN. This does not only mean that β_e does not change the set of β-SN terms, but also that we can actually use β_e with explicit substitution. In fact, explicit substitution, is an important topic of research and PSN is an important property for any λ-calculus extended with explicit substitution. In fact, lately, much research has been carried out ([BLR 95] and [KR 95]) in order to find systems of explicit substitution which are both confluent and have the PSN property (if M is β-SN then M is λ_s-SN where λ_s is the lambda calculus extended with explicit substitution). This is the reason for our interest in PSN of β_e (which is confluent by the way). After all, generalised reductions à la β_e have been extensively used as we saw in Section 1.1 for both theoretical and practical reasons. Furthermore, systems of explicit substitution have been the subject of much recent research. Both generalised reduction and explicit substitution are of practical importance and combining them both in one system may turn out to be very usef ul. Now, with PSN established we can study extending the λ-calculus with both explicit substitution and generalised reduction. This means that we can combine the advantages of the two different extensions in one system and we are investigating this line at the moment. Until the result of PSN of this paper, we were not sure which direction to take in combining both explicit substitution and generalised reduction in one system. We had established the following ($\lambda_{\beta_e s}$ stands for the lambda calculus extended with explicit substitution and generalised reduction and for reasons of uniformity, we write λ-SN for β-SN and λ_{β_e}-SN for β_e-SN):

(1) M is λ-SN $\Leftrightarrow$ M is λ_s-SN see [KR 95]

(2) M is λ_s-SN $\Leftrightarrow$ M is $\lambda_{\beta_e s}$-SN see [KR 96]

The proofs for (1) and (2) are similar. We had no idea however how to show either (3) or (4):

(3) M is λ-SN $\Leftrightarrow$ M is λ_{β_e}-SN

(4) M is λ_{β_e}-SN $\Leftrightarrow$ M is $\lambda_{\beta_e s}$-SN

With this paper, we establish (3) and hence we get (4) for free (because of the equivalences). Hence, one gets: M is λ-SN $\Leftrightarrow$ M is λ_{β_e}-SN $\Leftrightarrow$ M is $\lambda_{\beta_e s}$-SN $\Leftrightarrow$ M is λ_s-SN.

2 The formal machinery

We assume the reader familiar with the λ-calculus (whose terms are $\Lambda ::= \mathcal{V}|(\Lambda\Lambda)|(\lambda_{\mathcal{V}}.\Lambda)$), take terms modulo α-conversion and use the variable convention VC (as in [Bar 92]) which avoids any clash of variables. We use $x, y, z, x_1, x_2, \ldots$ and $M, N, P, Q, A, B, A_1, \ldots$ to range over $\mathcal{V}$ and Λ respectively. We assume the usual definition of substitution and use $FV(M)$ for the set of free variables of M. Because we need to see redexes (ordinary and generalised) we shall write terms in *item notation* (see [KN 96b] or [KN 95]). In this notation, λ_x is written as $[x]$ and (MN) is written $(N)M$ (note that following de Bruijn, we put the argument before the function). $[x]$ and (N) are called *items*. A sequence of items is called a *segment*. We use $I, I_1, \ldots$ to range over items and $S, S_1, S_2, \ldots$ to range over segments. A *well-balanced* segment (w.b for short) is defined as the empty segment or $(P)S_1[x]S_2$ where S_1 and S_2 are w.b. Note that the concatenation of w.b segments is a well-balanced segment.

One particular advantage of this notation is that redexes are more clear than in the usual notation. For example, γ_C of Subsection 1.1 becomes: $(Q)(P)[x][y]N \to (Q)[y](P)[x]N$ where it is clear that (P) matches $[x]$ and (Q) matches $[y]$. So, an ordinary redex starts with a ( ) adjacent to [ ]. A generalised redex starts with $(\,)S[\,]$ where S is w.b. When $S = \emptyset$, a generalised redex is an ordinary redex. In $(Q)(P)[x][y]N$, we say that (P), $[x]$, (Q) and $[y]$ are *partnered*, (P) is the *partner* of $[x]$ (or $[x]$ is the partner of (P)) and (Q) is the partner of $[y]$. (P) and $[x]$ are also said to be β-partnered whereas (Q) and $[y]$ are β_e-partnered. In general, we say that (P) (or $[x]$) is partnered in M if:

- $M \equiv (P)S[x]N$ where S is w.b (in this case (P) and $[x]$ are partners), or
- $M \equiv [y]N$ and (P) (or $[x]$) is partnered in N, or
- $M \equiv (N_1)N_2$ and (P) is either partnered in N_1 or in N_2.

We may also talk of β_I-, β_{eI}-, β_K-, β_{eK}-partnered items with the obvious meaning. Note that if $S_1(A)S_2[x]S_3$ is w.b where (A) and $[x]$ are partnered then S_2 and S_1S_3 are w.b.

If an item is not partnered in a term we say that it is *bachelor* (and may talk of β-, β_{eI}-, β_K-, β_{eK}-, β_I- and β_e-bachelor items). A segment consisting of bachelor items only is called bachelor. Note that a term will always be written as $I_1I_2\ldots I_nx$. Each I_i is said to be a *main*-item in M. A main item can of course have items inside it but these will not be main in M. For example, $((y)[x]x)[z]z$ has the main items $((y)[x]x)$ and $[z]$. The redex $((y)[x]x)[z]z$ is said to be a main-redex. The other redex $(y)[x]x$ is not main. The *weight* of a segment is defined to be the number of its main items. We write $[x := N]M$ instead of $M[x := N]$ which stands for substituting N for the free occurrences of x in M.

We assume the reader familiar with the basic machinery of reduction ([Bar 84], p. 50-59). In particular, if R is a binary relation $\subset \Lambda \times \Lambda$, and $(M, N) \in R$, we call M the R-redex and N the contractum of M. Given $R \subset \Lambda \times \Lambda$, we define $\to_R$ to be the least compatible relation containing R, $\twoheadrightarrow_R$ to be its reflexive transitive closure and $=_R$ to be its reflexive, symmetric and transitive closure. A term M is said to be in R-nf iff there is no N such that $M \to_R N$. M is said to have a R-nf, iff there is N in R-nf such that $M \twoheadrightarrow_R N$. We say M is R-normalising or is R-N iff M has a R-nf. We say that M is strongly R-normalising and write M is R-SN iff there is no infinite R-reduction path starting at M. We may use $M \twoheadrightarrow_R^+ N$ to indicate the existence of one or more steps from M to N and $M \twoheadrightarrow_R^n N$ to mean that there are n reduction steps. Ordinary β-, β_I- and β_K-reduction are defined as the reduction relations generated by the corresponding rules below:

(β)	$(N)[x]M \to [x := N]M$	
(β_I)	$(N)[x]M \to [x := N]M$	if $x \in FV(M)$
(β_K)	$(N)[x]M \to M$	if $x \notin FV(M)$

[dG 93] also uses

$$(\theta_K) \qquad (O)(N)[x]M \to (N)[x](O)M \qquad \text{if } x \notin FV(M)$$

Note that by VC, in θ_K, $x \notin FV(O)$. Moreover, de Groote moves (O) to the right of $(N)[x]$ so that it can eventually occur adjacent to its partner in M if it exists. De Groote establishes the following two results ($r \in \{\beta_I, \theta_K\}$):

(β_K-postponement for r)	If $M \to_{\beta_K} N \to_r O$ then $\exists P$ such that $M \twoheadrightarrow^+_{\beta_I\theta_K} P \twoheadrightarrow_{\beta_K} O$
(Conservation for $\beta + \theta$)	If M is $\beta_I\theta_K$-N then M is β-SN.

In this paper, we will improve both results. We will define a β_e-reduction relation (see Definition 2.1) whose β_{eI} and β_{eK} stand for its I and K-reductions. We shall show that:

(β_{eK}-postponement for β_e)	If $M \to_{\beta_{eK}} N \to_{\beta_{eI}} O$ then $\exists P$ such that $M \to_{\beta_{eI}} P \twoheadrightarrow^+_{\beta_{eK}} O$
(Conservation for β_e)	If M is β_{eI}-N then M is β_e-SN.

Definition 2.1 *(Generalised β-reduction β_e) We generalise β, β_I and β_K to the reduction relations generated by the corresponding rules of what follows:*

(β_e)	$(N)S[x]M \to S[x := N]M$	*if S is w.b*
(β_{eI})	$(N)S[x]M \to S[x := N]M$	*if S is w.b and $x \in FV(M)$*
(β_{eK})	$(N)S[x]M \to SM$	*if S is w.b and $x \notin FV(M)$*

Note that β_e is more generalised than the reduction relation introduced by combining de Groote's $\beta+\theta_K$. In fact, β_e is not restricted to K-redexes and one unique step can do the work of many in Groote's sense. For example, if $S \equiv (A_1)[x_1](A_2)[x_2]\ldots(A_n)[x_n]$ and all the redexes starting with $(A_1), (A_2), \ldots (A_n)$ are K-redexes in $S[x]M$, then $(N)S[x]M \to_{\beta_e} S[x := N]M$ iff $(N)S[x]M \twoheadrightarrow^n_{\theta K} S(N)[x]M \to_\beta S[x := N]M$.

Now, here is a basic lemma about terms:

Lemma 2.2

1. *Let $r \in \{\beta_e, \beta_{eI}, \beta_{eK}\}$. If (A) is r-bachelor in $(A)M$ then (B) is also r-bachelor in $(B)(A)M$.*
2. *If M is in β-nf, then $M \equiv [x_1][x_2]\ldots[x_n](A_1)(A_2)\ldots(A_m)z$ where $n \geq 0$, $m \geq 0$ and $\forall i$, $1 \leq i \leq m \Rightarrow A_i$ is in β-nf.*
3. *If $A \to_r A'$ then $SA \to_r SA'$ for any segment S and any reduction relation $r \in \{\beta, \beta_I, \beta_K, \beta_e, \beta_{eI}, \beta_{eK}\}$.*

Proof: *1. If (B) was r-partnered, then $(B)(A)M \equiv (B)(A)S[x]N$ where $(A)S$ is w.b (and hence $(A)S \equiv (A)S_1[y]S_2$ where S_1, S_2 are w.b) contradicting the fact that (A) is r-bachelor. 2. By induction on the structure of M. 3. By induction on the weight of S.* □

In order to show the Preservation of Strong Normalisation for β_e, we need a reduction strategy where a β_K-redex $(M)[x]N$ is contracted only if M is in β-nf. This strategy is actually the perpetual strategy (see [Bar 84] and [Reg 94]):

Definition 2.3 *We define the perpetual strategy F as follows:*

$F([x]M)$	$=$	$F(M)$	
$F((M)N)$	$=$	$F(N)$	*if $N \not\equiv [x]P$ and N is not in β-nf*
$F((M)N)$	$=$	$F(M)$	*if $N \not\equiv [x]P$ and N is in β-nf*
$F((M)[x]N)$	$=$	$(M)[x]N$	*if $x \in FV(N)$ or M is in β-nf*
$F((M)[x]N)$	$=$	$F(M)$	*if $x \notin FV(N)$ and M is not in β-nf*

We call perpetual reduction, the reduction associated with this strategy. When M β-reduces to N by contracting $F(M)$, we write, $M \to_F N$. This strategy has been shown in [Reg 94] to give the longest path for a SN term. It was moreover, shown in [Bar 84] that M is β-SN iff its perpetual reduction terminates. With the result of this paper, it will also be the case that M is β_e-SN iff its perpetual path terminates. The following lemma is informative about where F-reduction takes place in a term in the case of K-redexes:

Lemma 2.4 *If $M \to_F N$ where $F(M)$ is a β_K-redex, then one of the following holds:*

1. $M \equiv [x_1][x_2]\ldots[x_m](A_1)(A_2)\ldots(A_n)(A)[x]P$ *and*
$N \equiv [x_1][x_2]\ldots[x_m](A_1)(A_2)\ldots(A_n)P$
where $x \notin FV(P)$*,* A *is in* β*-nf,* $n \geq 0$ *and* $m \geq 0$*.*

2. $M \equiv [x_1][x_2]\ldots[x_m](A_1)(A_2)\ldots(A_n)(A)[x]P$ *and*
$N \equiv [x_1][x_2]\ldots[x_m](A_1)(A_2)\ldots(A_n)(A')[x]P$
where $x \notin FV(P)$*,* A *is not in* β*-nf,* $A \to_F A'$*,* $n \geq 0$ *and* $m \geq 0$*.*

3. $M \equiv [x_1][x_2]\ldots[x_m](A_1)(A_2)\ldots(A_n)(A)(B_1)(B_2)\ldots(B_r)z$ *and*
$N \equiv [x_1][x_2]\ldots[x_m](A_1)(A_2)\ldots(A_n)(A')(B_1)(B_2)\ldots(B_r)z$ *and*
A *is not in* β*-nf,* $A \to_F A'$*,* $n \geq 0$*,* $m \geq 0$ *and* $r \geq 0$ *and* $\forall i, 1 \leq i \leq r, B_i$ *is in* β*-nf.*

Proof: *By induction on* $M \to_F N$ *where* $F(M)$ *is a* β_K*-redex.*

- *Case* $(A)[x]P \to_F P$ *where* A *is in* β*-nf and* $x \notin FV(P)$*: this is form 1.*
- *Case* $(A)[x]P \to_F (A')[x]P$ *where* A *is not in* β*-nf and* $x \notin FV(P)$*: this is form 2.*
- *Case* $[x]M \to_F [x]M'$*, where* $M \to_F M'$*, use IH on* $M \to_F M'$*.*
- *Case* $(N)M \to_F (N)M'$*,* M *is not in* β*-nf and is not* $[y]Q$*, use IH on* $M \to_F M'$*.*
- *Case* $(A)B \to_F (A')B$*,* $B \not\equiv [y]Q$ *and* B *is in* β*-nf, by Lemma 2.2,* $B \equiv (B_1)\ldots(B_r)z$ *where all* B_i*s are in* β*-nf and we are done because this is form 3.* □

3 Postponement of β_{eK}-reduction

The proof of postponement is similar to that of de Groote. For us, however, we can get away with only one step β_{eI} reduction in the postponement lemma (Lemma 3.3). De Groote, had to have many steps in order to accommodate the slow process of moving an item ( ) next to its matching [ ] (see for example his proof of Lemma 11, (c) ii). We could also in Lemma 3.3 below, replace β_{eK} with ordinary β_K in $P \twoheadrightarrow^+_{\beta_{eK}} O$ but we won't bother doing this for this paper as it is not needed. Finally, note that in Lemma 3.3, $P \twoheadrightarrow^+_{\beta_{eK}} O$ and not $P \twoheadrightarrow_{\beta_{eK}} O$ nor $P \to_{\beta_{eK}} O$. This is due to Lemma 3.1.

Lemma 3.1 *If* $M \to_{\beta_{eK}} N$ *then*

1. $[x := M]P \twoheadrightarrow_{\beta_{eK}} [x := N]P$*.*
2. *if* $x \in FV(P)$ *then* $[x := M]P \twoheadrightarrow^+_{\beta_{eK}} [x := N]P$*.*

Proof: *Both 1 and 2 are by induction on the structure of* P*.* □

Lemma 3.2 *If* $M \to_{\beta_{eK}} N$ *then* $[x := P]M \to_{\beta_{eK}} [x := P]N$*.*
Proof: *By induction on the derivation of* $M \to_{\beta_{eK}} N$*.* □

Lemma 3.3 *If* $M \to_{\beta_{eK}} N \to_{\beta_{eI}} O$ *then* $\exists P$ *such that* $M \to_{\beta_{eI}} P \twoheadrightarrow^+_{\beta_{eK}} O$*.*
Proof: *By induction on the derivation of* $M \to_{\beta_{eK}} N$*.*

- *Case* $(A)S[x]B \to_{\beta_{eK}} SB$*,* S *w.b,* $x \notin FV(B)$*, check where in* SB *the* β_{eI}*-redex appears:*

- If $SB \to_{\beta_{eI}} SB' \equiv O$, then by compatibility, $(A)S[x]B \to_{\beta_{eI}} (A)S[x]B' \to_{\beta_{eK}} SB'$ as $x \notin FV(B')$.
  - If $S \equiv S_1(A_1)S_2[y]S_3$ where S_2 and S_1S_3 are w.b (note that $S = \emptyset$ is covered by the above case), and if $S_1(A_1)S_2[y]S_3B \to_{\beta_{eI}} S_1S_2\{[y := A_1]S_3\}[y := A_1]B$ then $(A)S_1(A_1)S_2[y]S_3[x]B \to_{\beta_{eI}} (A)S_1S_2\{[y := A_1]S_3\}[x]\{[y := A_1]B\} \to_{\beta_{eK}} S_1S_2\{[y := A_1]S_3\}\{[y := A_1]B\}$ as $x \notin FV([y := A_1]B)$ due to VC.
  - If $S \equiv S_1(A_1)S_2[y]S_3$, S_2, S_1S_3 are w.b and $S_1(A_1)S_2[y]S_3B \to_{\beta_{eI}} S_1(A_1')S_2[y]S_3B$ then $(A)S_1(A_1)S_2[y]S_3[x]B \to_{\beta_{eI}} (A)S_1(A_1')S_2[y]S_3[x]B \to_{\beta_{eK}} S_1(A_1')S_2[y]S_3B$.

- Case $[x]M \to_{\beta_{eK}} [x]N \to_{\beta_{eI}} O$, then $O \equiv [x]Q$ use IH on $M \to_{\beta_{eK}} N \to_{\beta_{eI}} Q$.

- Case $(A)B \to_{\beta_{eK}} (A')B \to_{\beta_{eI}} O$, we investigate how $(A')B \to_{\beta_{eI}} O$.
  - If $O \equiv (A')B'$ where $B \to_{\beta_{eI}} B'$, then $(A)B \to_{\beta_{eI}} (A)B' \to_{\beta_{eK}} (A')B'$.
  - If $O \equiv (A'')B$ where $A' \to_{\beta_{eI}} A''$, Use IH on $A \to_{\beta_{eK}} A' \to_{\beta_{eI}} A''$ and compatibility
  - If $(A')B \equiv (A')S[x]B_1 \to_{\beta_{eI}} S[x := A']B_1$, then
    $(A)B \equiv (A)S[x]B_1 \to_{\beta_{eI}} S[x := A]B_1 \twoheadrightarrow^+_{\beta_{eK}} S[x := A']B_1$ by Lemma 3.1.

- Case $(A)B \to_{\beta_{eK}} (A)B' \to_{\beta_{eI}} O$ then
  - If $O \equiv (A)B''$ where $B' \to_{\beta_{eI}} B''$ then use IH on $B \to_{\beta_{eK}} B' \to_{\beta_{eI}} B''$ and derive $(A)B \to_{\beta_{eI}} (A)P \twoheadrightarrow^+_{\beta_{eK}} (A)B''$.
  - If $O \equiv (A')B'$ where $A \to_{\beta_{eI}} A'$ then $(A)B \to_{\beta_{eI}} (A')B \to_{\beta_{eK}} (A')B'$.
  - If $B' \equiv S[x]C$, S is w.b, and $O \equiv S[x := A]C$. I.e. $(A)B \to_{\beta_{eK}} (A)S[x]C \to_{\beta_{eI}} S[x := A]C$.
    Case $B \equiv S[x]C_1$ and $C_1 \to_{\beta_{eK}} C$, $(A)S[x]C_1 \to_{\beta_{eI}} S[x := A]C_1 \to_{\beta_{eK}} S[x := A]C$ by Lemma 3.2.
    Case $B \equiv S_1[x]C$ and the β_{eK}-redex is in S_1, i.e. $(A)S_1[x]C \to_{\beta_{eK}} (A)S[x]C \to_{\beta_{eI}} S[x := A]C$. Now, $(A)S_1[x]C \to_{\beta_{eI}} S_1[x := A]C \to_{\beta_{eK}} S[x := A]C$ by VC. □

4 The generalised conservation for β_e

The set ${}^{I\!\!N}\Lambda$ of labelled λ-terms is inductively defined as follows:

1. $n \in I\!\!N, x \in \mathcal{V} \Rightarrow {}^n x \in {}^{I\!\!N}\Lambda$.
2. $n \in I\!\!N, x \in \mathcal{V}, M \in {}^{I\!\!N}\Lambda \Rightarrow {}^n[x]M \in {}^{I\!\!N}\Lambda$.
3. $n \in I\!\!N, M, N \in {}^{I\!\!N}\Lambda \Rightarrow {}^n(M)N \in {}^{I\!\!N}\Lambda$

We take $M, N, O, A, B, \ldots$ to range over labelled λ-terms. We use ${}^nM$ to stress that the outermost label of a λ-term M is n. Hence, M and ${}^nM$ stand for the same labelled λ-term. We write ${}^{+m}M$ for the labelled λ-term obtained by adding m to the outermost label of a labelled λ-term M. Hence if the outermost label of M is n then ${}^{+m}M$ denotes ${}^{n+m}M$.

For $M \in {}^{I\!\!N}\Lambda$, we write $|M|$ for the (unlabelled) λ-term in Λ obtained by erasing all labels in M. Moreover, if $M \in \Lambda$, we identify M with M' in ${}^{I\!\!N}\Lambda$ such that $|M'| \equiv M$ and all labels in M' are 0. Hence, $\Lambda \subset {}^{I\!\!N}\Lambda$.

Labels are used as counters to record the number of contracted redexes when reducing a term. We use in this section, the notations and techniques of de Groote adapted however to our generalised reduction. Basically the idea is as follows: we introduce a labelled reduction relation $\to_{\beta^+_{eI}}$ which we prove Church Rosser. $\to_{\beta^+_{eI}}$ is shown CR by showing that a related reduction relation $\to_1$ is CR. Hence, if a labelled term M has a β^+_{eI}-nf, it must be unique. We then introduce the notion of *weight* of a term M, $\Theta[M]$, which is used to limit the length of β^+_{eI}-reductions starting at normalising terms. That is, the length of any sequence of β^+_{eI}-reductions starting at a normalising term M is bounded by $\Theta[M'] - \Theta[M]$ where M' is the (unique) β^+_{eI}-nf of M. This implies that any β^+_{eI}-N term is β^+_{eI}-SN. This will be extended to β_{eI} by showing that any β_{eI}-N term is β_{eI}-SN. Next we show that if M is β_{eI}-N then it is β_e-SN by using the fact that M is β_{eI}-SN, postponement and that there can only be a finite β_{eK}-redexes.

Here is the definition of substitution on labelled terms and a basic lemma about substitution:

Definition 4.1 *Let $M, N \in {}^N\Lambda$. $[x := N]M$ is defined as follows:*

$$\begin{array}{lll}
[x := {}^nN]^m x & \equiv & {}^{n+m}N \\
[x := {}^nN]^m y & \equiv & {}^m y \qquad \text{if } x \not\equiv y \\
[x := {}^nN]^m (P)Q & \equiv & {}^m([x := {}^nN]P)[x := {}^nN]Q \\
[x := {}^nN]^m [y]M & \equiv & {}^m[y][x := {}^nN]M
\end{array}$$

Lemma 4.2 *Let $P, C, D \in {}^N\Lambda$, $x \not\equiv y$ and $y \notin FV(P)$. The following holds:*

$[x := {}^nP][y := {}^mC]^oD \equiv [y := [x := {}^nP]^mC][x := {}^nP]^oD$. Moreover, both sides have the label $n+m+o$ if $D \equiv y \wedge C \equiv x$, $m+o$ if $D \equiv y \wedge C \not\equiv x$, $n+o$ if $D \equiv x$, and o otherwise. Furthermore, ${}^{+i}[y := {}^mC]^oD \equiv [y := {}^mC]^{+i}D$.

Proof: *By induction on the structure of D leaving to the reader the easy ${}^{+i}[y := {}^mC]^oD \equiv [y := {}^mC]^{+i}D$ (which is proven by cases on the structure of D).*

- *$D \equiv {}^oy$ then lhs $\equiv [x := {}^nP]^{m+o}C$, rhs $\equiv [y := [x := {}^nP]^mC]^oy$. If $C \equiv x$ then both sides are ${}^{n+m+o}P$. If $C \not\equiv x$ then both sides have $m+o$ as labels and are ${}^{m+o}[x := P]C$.*
- *$D \equiv {}^ox$ then both sides are ${}^{o+n}P$ as $y \notin FV(P)$.*
- *$D \equiv {}^oz$, $z \not\equiv x$ and $z \not\equiv y$, then both sides are ${}^oz$.*
- *$D \equiv {}^o[z]A$ then use IH.*
- *$D \equiv {}^o(A)B$ then*
  $[x := {}^nP][y := {}^mC]^o(A)B \equiv$
  ${}^o([x := {}^nP][y := {}^mC]A)[x := {}^nP][y := {}^mC]B \equiv^{IH}$
  ${}^o([y := [x := {}^nP]^mC][x := {}^nP]A)[y := [x := {}^nP]^mC][x := {}^nP]B \equiv$
  $[y := [x := {}^nP]^mC][x := {}^nP]^o(A)B.$ □

Now we define $\to_{\beta^+_{eI}}$ which will be used to show conservation.

Definition 4.3 *$M \to_{\beta^+_{eI}} N$ is defined inductively as follows:*

1. *${}^n({}^iN)S^o[x]^jM \to_{\beta^+_{eI}} {}^{+n+o+1}S[x := {}^iN]^jM$ if $x \in FV(M)$, S, w.b.*
2. *If $M \to_{\beta^+_{eI}} N$ then ${}^n[x]M \to_{\beta^+_{eI}} {}^n[x]N$*

3. *If* $M \to_{\beta^+_{eI}} N$ *then* ${}^n(M)P \to_{\beta^+_{eI}} {}^n(N)P$ *and* ${}^n(P)M \to_{\beta^+_{eI}} {}^n(P)N$

$\twoheadrightarrow_{\beta^+_{eI}}$ *is defined as the transitive reflexive closure of* $\to_{\beta^+_{eI}}$.

We define $\to_1$ for which CR is easier to show than for $\to_{\beta^+_{eI}}$.

Definition 4.4 *$M \to_1 N$ is defined inductively as follows:*

1. $M \to_1 M$
2. *If* $M \to_1 N$ *then* ${}^n[x]M \to_1 {}^n[x]N$
3. *If* $M \to_1 O$ *and* $N \to_1 P$ *then* ${}^n(M)N \to_1 {}^n(O)P$
4. *If* $S^p[x]M \to_1 S'^q[x]O$, $N \to_1 P$, S, S' *w.b, and* $x \in FV(M)$ *then* ${}^n(N)S^p[x]M \to_1 {}^{+n+q+1}S'[x := P]O$.

$\twoheadrightarrow_1$ *is defined as the transitive reflexive closure of* $\to_1$.

The following lemma shows that labels can be increased for both $\to_1$ and $\to_{\beta^+_{eI}}$.

Lemma 4.5 *Let* $M, N \in {}^N\Lambda$.

1. $M \to_1 N$ *then* ${}^{+n}M \to_1 {}^{+n}N$.
2. $M \to_{\beta^+_{eI}} N$ *then* ${}^{+n}M \to_{\beta^+_{eI}} {}^{+n}N$.

Proof: *We only show 1 by induction on the derivation $M \to_1 N$. 2 is similar and is by induction on the derivation $M \to_{\beta^+_{eI}} N$.*

1. ${}^{+n}M \to_1 {}^{+n}M$ *by def. of* $\to_1$.
2. *If* ${}^m[x]M \to_1 {}^m[x]N$ *where* $M \to_1 N$ *then by def. of* $\to_1$, ${}^{m+n}[x]M \to_1 {}^{m+n}[x]N$.
3. *If* ${}^m(M)N \to_1 {}^m(O)P$, $M \to_1 O$, $N \to_1 P$ *then by def. of* $\to_1$, ${}^{m+n}(M)N \to_1 {}^{m+n}(O)P$.
4. *If* ${}^m(N)S^p[x]M \to_1 {}^{+m+q+1}S'[x := P]O$, $N \to_1 P$, $S^p[x]M \to_1 S'^q[x]O$, S, S', *w.b,* $x \in FV(M)$ *then by def. of* $\to_1$, ${}^{n+m}(N)S^p[x]M \to_1 {}^{+n+m+q+1}S'[x := P]O$. □

The following lemma shows that $\to_1$ and $\to_{\beta^+_{eI}}$ close under substitution.

Lemma 4.6 *Let* $M, N, O \in {}^N\Lambda$.

1. *If* $M \to_1 N$, *then* $[x := M]^m O \to_1 [x := N]^m O$.
2. *If* $M \to_{\beta^+_{eI}} N$, *then* $[x := M]^m O \twoheadrightarrow_{\beta^+_{eI}} [x := N]^m O$.

Proof: *1 and 2 are similar and are by induction on the structure of O. We only show 1.*

- $[x := M]^m x \equiv {}^{+m}M \to_1 {}^{+m}N \equiv [x := N]^m x$ *using Lemma 4.5.*
- $[x := M]^m y \equiv {}^m y \to_1 {}^m y \equiv [x := N]^m y$ *for* $y \not\equiv x$
- $[x := M]^m[y]O \equiv {}^m[y][x := M]O \to_1{}^{IH}\ {}^m[y][x := N]O \equiv [x := N]^m[y]O$

- $[x:=M]^m(P)Q \equiv {}^m([x:=M]P)[x:=M]Q \to_1{}^{IH}$
  ${}^m([x:=N]P)[x:=N]Q \equiv [x:=N]^m(P)Q$ □

Lemma 4.7 *Let $M, N, P, O \in {}^N\Lambda$.*

1. *If $M \to_1 N$ and $O \to_1 P$ then $[x:=O]M \to_1 [x:=P]N$.*
2. *If $M \to_{\beta^+_{eI}} N$ and $O \to_{\beta^+_{eI}} P$ then $[x:=O]M \twoheadrightarrow_{\beta^+_{eI}} [x:=P]N$.*

Proof: *We only prove 1 by induction on the derivation $M \to_1 N$. 2 is similar and is by induction on the derivation $M \to_{\beta^+_{eI}} N$.*

- *Case $M \to_1 M$ then by Lemma 4.6, $[x:=O]M \to_1 [x:=P]M$.*
- *Case ${}^m[y]M \to_1 {}^m[y]N$ where $M \to_1 N$, then by IH, $[x:=O]M \to_1 [x:=P]N$ and hence $[x:=O]^m[y]M \equiv {}^m[y][x:=O]M \to_1 {}^m[y][x:=P]N \equiv [x:=P]^m[y]N$.*
- *Case ${}^m(A)B \to_1 {}^m(C)D$ where $A \to_1 C$ and $B \to_1 D$ then by IH, $[x:=O]A \to_1 [x:=P]C$ and $[x:=O]B \to_1 [x:=P]D$. So, $[x:=O]^m(A)B \equiv {}^m([x:=O]A)[x:=O]B$ $\to_1 {}^m([x:=P]C)[x:=P]D \equiv [x:=P]^m(C)D$.*
- *Case ${}^n(A)S^p[y]B \to_1 {}^{+n+q+1}S'[y:=C]D$ where $A \to_1 C$, $S^p[y]B \to_1 S'^q[y]D$, S, S' w.b, and $y \in FV(B)$, then by IH, $[x:=O]A \to_1 [x:=P]C$ and*
  $\{[x:=O]S\}^p[y][x:=O]B \equiv [x:=O]S^p[y]B \to_1$
  $[x:=P]S'^q[y]D \equiv \{[x:=P]S'\}^q[y][x:=P]D$. *Now,*
  $[x:=O]^n(A)S^p[y]B \equiv {}^n([x:=O]A)[x:=O]S^p[y]B \to_1^{y\in FV([x:=O]B)}$
  ${}^{+n+q+1}\{[x:=P]S'\}[y:=[x:=P]C][x:=P]D \equiv^{y\notin FV(P), Lemma\ 4.2}$
  ${}^{+n+q+1}\{[x:=P]S'\}[x:=P][y:=C]D \equiv^{Lemma\ 4.2} [x:=P]^{+n+q+1}S'[y:=C]D$ □

Here is the relationship between $\to_1$ and $\to_{\beta^+_{eI}}$:

Lemma 4.8 *$M \twoheadrightarrow_1 N$ iff $M \twoheadrightarrow_{\beta^+_{eI}} N$.*

Proof: *⇒) By induction on the derivation of $M \to_1 N$ show that $M \to_1 N \Rightarrow M \twoheadrightarrow_{\beta^+_{eI}} N$.*

- *If $M \to_1 M$ then obvious. If ${}^n[x]M \to_1 {}^n[x]N$ or ${}^n(M)P \to_1 {}^n(N)Q$ where $M \to_1 N$ and $P \to_1 Q$ use IH.*
- *If ${}^n(N)S^p[x]M \to_1 {}^{+n+q+1}S'[x:=P]O$ where S, S' w.b, $x \in FV(M)$, $S^p[x]M \to_1 S'^q[x]O$ and $N \to_1 P$ then by IH, $S^p[x]M \twoheadrightarrow_{\beta^+_{eI}} S'^q[x]O$ and $N \twoheadrightarrow_{\beta^+_{eI}} P$.*
  Hence, ${}^n(N)S^p[x]M \twoheadrightarrow_{\beta^+_{eI}} {}^n(P)S'^q[x]O \to_{\beta^+_{eI}} {}^{+n+q+1}S'[x:=P]O$.

⇐) By induction on the derivation $M \to_{\beta^+_{eI}} N$, show that $M \to_{\beta^+_{eI}} N \Rightarrow M \to_1 N$.

- *If ${}^n(N)S^p[x]M \to_{\beta^+_{eI}} {}^{+n+p+1}S[x:=N]M$ where S w.b, $x \in FV(M)$, then as $N \to_1 N$ and $S^p[x]M \to_1 S^p[x]M$, we are done.*
- *If ${}^n[x]M \to_{\beta^+_{eI}} {}^n[x]N$ or ${}^n(M)P \to_{\beta^+_{eI}} {}^n(N)P$ or ${}^n(P)M \to_{\beta^+_{eI}} {}^n(P)N$ where $M \to_{\beta^+_{eI}} N$ use IH and $P \to_1 P$.* □

The following two lemmas enable us to establish that $\to_1$ is CR.

Lemma 4.9 *If S, S' w.b, none of the binding variables of $S[x]$ occurs free in N, none of the binding variables of $S'[x]$ occurs free in P, none of the binding variables of N are free in M and none of the binding variables of P are free in O, $S^p[x]M \to_1 S'^q[x]O$ and $N \to_1 P$ then ${}^{+p}S[x := N]M \to_1 {}^{+q}S'[x := P]O$.*

Proof: *Note that if* $\texttt{weight}(S) = \texttt{weight}(S')$ *then $p = q$, $M \to_1 O$ and if (A_i) and (A'_i) are the ith main application items of S and S' respectively, then $A_i \to_1 A'_i$. Hence the result is shown by Lemmas 4.5 and 4.7 and the def. of $\to_1$.*
If $\texttt{weight}(S) > \texttt{weight}(S')$*, then we prove the lemma by induction on* $\texttt{weight}(S)$.

- *If $S \equiv {}^n(A)^o[y]$ then $S' \equiv \emptyset$, $q = p + n + o + 1$, $O \equiv [y := A']M'$ where $A \to_1 A'$ and $M \to_1 M'$. Hence by Lemma 4.7 ${}^o[y][x := N]M \to_1 {}^o[y][x := P]M'$ and so by def. of $\to_1$, ${}^{n+p}(A)^o[y][x := N]M \to_1 {}^{+n+p+o+1}[y := A'][x := P]M' \equiv^{Lemma\ 4.2}$ ${}^{+n+p+o+1}[x := P][y := A']M' \equiv {}^{+n+p+o+1}[x := P]O$ and we are done.*

- *The inductive case is long but straightforward.* □

Lemma 4.10 *Let $M, N, O \in {}^N\Lambda$ such that $M \to_1 N$ and $M \to_1 O$ then $\exists P \in {}^N\Lambda$ such that $N \to_1 P$ and $O \to_1 P$.*

Proof: *By induction on the derivation of $M \to_1 N$.*

1. *$M \to_1 M$ then $P \equiv O$.*

2. *${}^n[x]M_1 \to_1 {}^n[x]N_1$ where $M_1 \to_1 N_1$, then $O \equiv {}^n[x]O_1$ and $M_1 \to_1 O_1$. Now use IH.*

3. *${}^n(M_1)M_2 \to_1 {}^n(N_1)N_2$ where $M_1 \to_1 N_1$ and $M_2 \to_1 N_2$, then*

   (a) *If $O \equiv {}^n(O_1)O_2$, $M_1 \to_1 O_1$ and $M_2 \to_1 O_2$ then by IH, $\exists P_1, P_2$ such that $N_1 \to_1 P_1, O_1 \to_1 P_1, N_2 \to_1 P_2$ and $O_2 \to_1 P_2$. Hence ${}^n(N_1)N_2 \to_1 {}^n(P_1)P_2$ and ${}^n(O_1)O_2 \to_1 {}^n(P_1)P_2$.*

   (b) *If $M_2 \equiv S^p[x]M_3$, $O \equiv {}^{+n+q+1}S'[x := O_1]O_2$ where $M_1 \to_1 O_1$, $S^p[x]M_3 \to_1 S'^q[x]O_2$, $x \in FV(M_3)$, S, S' w.b, then as S is w.b, $N_2 \equiv S''^o[x]N_3$, S'' w.b (which may or may not be empty) and $x \in FV(N_3)$. Now, by IH, $\exists P_1, P_2, S'''$ w.b, such that, $N_1 \to_1 P_1, O_1 \to_1 P_1, S''^o[x]N_3 \to_1 S'''^r[x]P_2$ and $S'^q[x]O_2 \to_1 S'''^r[x]P_2$. Hence, by Lemmas 4.5 and 4.9, ${}^{+n+q+1}S'[x := O_1]O_2 \to_1 {}^{+n+r+1}S'''[x := P_1]P_2$. Note that all the preconditions of Lemma 4.9 hold due to the variable convention VC. Now, as $S''^o[x]N_3 \to_1 S'''^r[x]P_2$, $x \in FV(N_3)$ and $N_1 \to_1 P_1$, we get by def. of $\to_1$, ${}^n(N_1)S''^o[x]N_3 \to_1 {}^{+n+r+1}[x := P_1]P_2$.*

4. *Case ${}^n(M_1)S^p[x]M_2 \to_1 {}^{+n+q+1}S'[x := N_1]N_2$, $M_1 \to_1 N_1$, $S^p[x]M_2 \to_1 S'^q[x]N_2$, S, S' w.b, $x \in FV(M_2)$:*

   (a) *Case $O \equiv {}^n(O_1)O_2$ similar to case 3, (b).*

   (b) *Case $O \equiv {}^{+n+o+1}S''[x := O_1]O_2$ where $M_1 \to_1 O_1$ and $S^p[x]M_2 \to_1 S''^o[x]O_2$, S'' w.b, then by IH, $\exists P_1, P_2, S'''$ w.b, such that $N_1 \to_1 P_1, O_1 \to_1 P_1, S'^q[x]N_2 \to_1 S'''^r[x]P_2, S''^o[x]O_2 \to_1 S'''^r[x]P_2$ and by Lemmas 4.5 and 4.9 (again here, all the preconditions of Lemma 4.9 hold due to the variable convention VC):*
   ${}^{+n+q+1}S'[x := N_1]N_2 \to_1 {}^{+n+r+1}S'''[x := P_1]P_2$ *and*
   ${}^{+n+o+1}S''[x := O_1]O_2 \to_1 {}^{+n+r+1}S'''[x := P_1]P_2$. □

Corollary 4.11 *(Church Rosser of $\rightarrow_1$) Let $M, N, O \in {}^N\Lambda$ such that $M \twoheadrightarrow_1 N$ and $M \twoheadrightarrow_1 O$ then $\exists P \in {}^N\Lambda$ such that $N \twoheadrightarrow_1 P$ and $O \twoheadrightarrow_1 P$.* □

Now, the first part of this section (CR of $\rightarrow_{\beta^+_{eI}}$) is done:

Lemma 4.12 *(Church Rosser of $\rightarrow_{\beta^+_{eI}}$) Let $M, N, O \in {}^N\Lambda$ such that $M \twoheadrightarrow_{\beta^+_{eI}} N$ and $M \twoheadrightarrow_{\beta^+_{eI}} O$ then $\exists P \in {}^N\Lambda$ such that $N \twoheadrightarrow_{\beta^+_{eI}} P$ and $O \twoheadrightarrow_{\beta^+_{eI}} P$.*

Proof: *By Corollary 4.11 and Lemma 4.8.* □

In order to show Lemma 4.16, we introduce the following definition:

Definition 4.13 *The weight $\Theta[M]$ of a labelled λ-term M is defined as follows:*

$$\begin{array}{lcl}
\Theta[{}^n x] & = & n \\
\Theta[{}^n [y]M] & = & n + \Theta[M] \\
\Theta[{}^n (M)N] & = & n + \Theta[M] + \Theta[N]
\end{array}$$

Lemma 4.14 *If $x \in FV(M)$ then $\Theta[[x := N]M] \geq \Theta[M] + \Theta[N]$.*

Proof: *By induction on the structure of M showing first that $\Theta[{}^{+m} M] = m + \Theta[M]$.* □

Lemma 4.15 *Let $M, N \in {}^N\Lambda$ and $M \twoheadrightarrow^+_{\beta^+_{eI}} N$ then $\Theta[M] < \Theta[N]$.*

Proof: *By induction on the derivation $M \twoheadrightarrow^+_{\beta^+_{eI}} N$ using Lemma 4.14.* □

Now, β^+_{eI}-N and β^+_{eI}-SN are the same:

Lemma 4.16 *If M is β^+_{eI}-N then M is β^+_{eI}-SN.*

Proof: *Since M is β^+_{eI}-N, and since β^+_{eI} is Church Rosser by Lemma 4.12, then M has a unique β^+_{eI}-nf M'. According to Lemma 4.15, the length of any sequence of β^+_{eI}-reduction starting at M is bounded by $\Theta[M'] - \Theta[M]$.* □

Here is the relationship between $\rightarrow_{\beta_{eI}}$ and $\rightarrow_{\beta^+_{eI}}$:

Lemma 4.17 *Let $M, N \in \Lambda$ such that $M \rightarrow_{\beta_{eI}} N$, then there exist $M', N' \in {}^N\Lambda$ such that $|M'| \equiv M, |N'| \equiv N$ and $M' \rightarrow_{\beta^+_{eI}} N'$. Furthermore, if N is in β_{eI}-nf then N' is in β^+_{eI}-nf.*

Proof: *This is easy. Just put the right labels on M and N obtaining M', N' such that $M' \rightarrow_{\beta^+_{eI}} N'$.* □

Now, we generalise Lemma 4.16 to $\rightarrow_{\beta_{eI}}$.

Theorem 4.18 *If M is β_{eI}-N then M is β_{eI}-SN.*

Proof: *M β_{eI}-N $\Rightarrow^{Lemma\ 4.17}$ M β^+_{eI}-N $\Rightarrow^{Lemma\ 4.16}$ M β^+_{eI}-SN $\Rightarrow$ M β_{eI}-SN (otherwise there exists an infinite β^+_{eI}-path).* □

Finally, conservation results from the above theorem and postponement of K-contractions.

Theorem 4.19 *(Conservation) If M is β_{eI}-N then M is β_e-SN.*

Proof: *If M is not β_e-SN then there is an infinite β_e-path starting at M. But by postponement of β_{eK} redexes, and by the fact that there can only be a finite β_{eK}-contractions, there must be an infinite β_{eI}-path. But M is β_{eI}-N and so it is β_{eI}-SN by theorem 4.18. Contradiction.* □

5 Preservation of Strong Normalisation

In this section, following Theorem 4.18, we interchange β_{eI}-SN and β_{eI}-N at liberty. We shall show PSN of β_e. Note that this is not straightforward. Take for example the following derivation:

$$M\ \beta\text{-SN} \Rightarrow M\ \beta\text{-N} \Rightarrow M\ \beta_{eI}\text{-N} \Rightarrow M\ \beta_e\text{-SN}$$

This is incorrect because M β-N $\not\Rightarrow$ M β_{eI}-N. For example, $(\lambda_x.y)\Omega$ is β-N but not β_{eI}-N for $\Omega \equiv (\lambda_z.zz)(\lambda_z.zz)$. In fact, showing PSN was not easy to establish until it was realised that a reduction strategy whose inverse preserves β_{eI}-normalisation was needed. It turned out that this is the perpetual strategy. Once this was established, PSN was in sight. Take M that is β-SN. Then $M \twoheadrightarrow_F N$ where N is the β-nf of M and $\twoheadrightarrow_F$ is the perpetual strategy. As N is in β-nf, then N is β_{eI}-N. But the inverse of $\twoheadrightarrow_F$ preserves β_{eI}-N. Hence, M is β_{eI}-N and by conservation, M is β_e-SN.

In order to establish that the inverse of $\twoheadrightarrow_F$ preserves β_{eI}-normalisation (Theorem 5.5), we need the following three lemmas which will be combined with the three forms of perpetual reduction for K-redexes as in Lemma 2.4.

Lemma 5.1 *If $(A_1)\ldots(A_n)(A)[x]P$ has β_{eI}-nf, $x \notin FV(P)$, then its β_{eI}-nf is of the form $(B_1)\ldots(B_j)(A')[x]Q$ where A' is the β_{eI}-nf of A, $0 \leq j \leq n$, B_j is the β_{eI}-nf of some A_i for $1 \leq i \leq n$. Moreover, $(A_1)\ldots(A_n)P$ has $(B_1)\ldots(B_j)Q$ as its β_{eI}-nf.*

Proof: *By induction on $n \geq 0$.*

- *$n = 0$, the β_{eI}-nf is $(A')[x]Q$ where Q is the β_{eI}-nf of P.*
- *Assume the property holds for $n \geq 0$. As $(A_1)\ldots(A_n)(A_{n+1})(A)[x]P$ has β_{eI}-nf, then it is β_{eI}-SN and so $(A_2)\ldots(A_n)(A_{n+1})(A)[x]P$ and A_1 have β_{eI}-nf. Call the β_{eI}-nf of A_1, A'_1. Now, by IH, $(B_1)\ldots(B_j)(A')[x]Q$ is the β_{eI}-nf of $(A_2)\ldots(A_n)(A_{n+1})(A)[x]P$ and $(B_1)\ldots(B_j)Q$ is the β_{eI}-nf of $(A_2)\ldots(A_n)(A_{n+1})P$.*
  - *If (A'_1) is β_{eI}-bachelor in $(A'_1)(B_1)\ldots(B_j)(A')[x]Q$ (and so in $(A'_1)(B_1)\ldots(B_j)Q$), then $(A'_1)(B_1)\ldots(B_j)(A')[x]Q$ and $(A'_1)(B_1)\ldots(B_j)Q$ are the β_{eI}-nfs required.*
  - *If (A'_1) is β_{eI}-partnered in $(A'_1)(B_1)\ldots(B_j)(A')[x]Q$ then all $(B_1),\ldots(B_j)$ start β_{eK}-redexes and $Q \equiv [x_j]\ldots[x_1][y]R$.*
    Now, $(A_1)\ldots(A_n)(A_{n+1})(A)[x]P \twoheadrightarrow_{\beta_{eI}} (A'_1)(B_1)\ldots(B_j)(A')[x][x_j]\ldots[x_1][y]R \rightarrow_{\beta_{eI}} (B_1)\ldots(B_j)(A')[x][x_j]\ldots[x_1][y := A'_1]R \twoheadrightarrow_{\beta_{eI}} (B_1)\ldots(B_j)(A')[x][x_j]\ldots[x_1]B$ for B the β_{eI}-nf of $[y := A'_1]R$.
    Moreover, $(A_1)\ldots(A_n)(A_{n+1})P \twoheadrightarrow_{\beta_{eI}} (A'_1)(B_1)\ldots(B_j)[x_j]\ldots[x_1][y]R \rightarrow_{\beta_{eI}} (B_1)\ldots(B_j)[x_j]\ldots[x_1][y := A'_1]R \twoheadrightarrow_{\beta_{eI}} (B_1)\ldots(B_j)[x_j]\ldots[x_1]B$. Now, we are done (note that $B_1,\ldots B_j$ start β_{eK}-redexes). □

Lemma 5.2 *If $(A_1)\ldots(A_n)P$ and A have β_{eI}-nf, $x \notin FV((A_1)\ldots(A_n)(A)P)$, then: $(A_1)\ldots(A_n)(A)[x]P$ has β_{eI}-nf.*

Proof: *By induction on $n \geq 0$.*

- *Case $n = 0$, P and A have P' and A' as β_{eI}-nfs, then $(A)[x]P$ has $(A')[x]P'$ as β_{eI}-nf.*

- *Assume the property holds for* $n \geq 0$. *Let* $(A_1)\ldots(A_n)(A_{n+1})P$ *have* β_{eI}*-nf, hence it is* β_{eI}*-SN and so* $(A_2)\ldots(A_{n+1})P$ *has* β_{eI}*-nf and* A_1 *has* A'_1 *as* β_{eI}*-nf. By IH,* $(A_2)\ldots(A_{n+1})(A)[x]P$ *has* β_{eI}*-nf which is by Lemma 5.1,* $M \equiv (B_1)\ldots(B_j)(A')[x]Q$ *and* $(A_2)\ldots(A_{n+1})P$ *has* $(B_1)\ldots(B_j)Q$ *as its* β_{eI}*-nf.*

  - *If* (A'_1) *is* β_{eI}*-bachelor in* $(A'_1)M$ *then* $(A'_1)M$ *is the* β_{eI}*-nf of* $(A_1)\ldots(A_{n+1})(A)[x]P$.
  - *If* (A'_1) *is* β_{eI}*-partnered in* $(A'_1)M$ *then* $Q \equiv [x_j]\ldots[x_1][y]R$. *Now,* $(A_1)\ldots(A_n)(A_{n+1})P \twoheadrightarrow_{\beta_{eI}} (A'_1)(B_1)\ldots(B_j)[x_j]\ldots[x_1][y]R \rightarrow_{\beta_{eI}}$ $(B_1)\ldots(B_j)[x_j]\ldots[x_1][y := A'_1]R$. *Hence* $[y := A'_1]R$ *is* β_{eI}*-SN as* $(A_1)\ldots(A_{n+1})P$ *is. Let* B *be the* β_{eI}*-nf of* $[y := A'_1]R$. *Now,* $(A_1)\ldots(A_{n+1})(A)[x]P \twoheadrightarrow_{\beta_{eI}} (A'_1)(B_1)\ldots(B_j)(A')[x][x_j]\ldots[x_1][y]R \rightarrow_{\beta_{eI}}$ $(B_1)\ldots(B_j)(A')[x][x_j]\ldots[x_1][y := A'_1]R \rightarrow_{\beta_{eI}} (B_1)\ldots(B_j)(A')[x][x_j]\ldots[x_1]B$ *which is in* β_{eI}*-nf.* □

Lemma 5.3 *If* $A_i \rightarrow_{\beta_K} B_i$, $(A_1)\ldots(A_{i-1})(B_i)(A_{i+1})\ldots(A_n)z$ *and* A_i *have* β_{eI}*-nf then* $(A_1)\ldots(A_{i-1})(A_i)(A_{i+1})\ldots(A_n)z$ *has* β_{eI}*-nf.*

Proof: $A_1,\ldots A_{i-1}, A_i, A_{i+1},\ldots A_n$ *all have* β_{eI}*-nf,* $A'_1,\ldots A'_{i-1}, A'_i, A'_{i+1},\ldots A'_n$. *Hence,* $(A_1)\ldots(A_{i-1})(A_i)(A_{i+1})\ldots(A_n)z \twoheadrightarrow_{\beta_{eI}} (A'_1)\ldots(A'_{i-1})(A'_i)(A'_{i+1})\ldots(A'_n)z$ *in* β_{eI}*-nf.* □

Lemma 5.4 *If* $M \rightarrow_F N$ *using a* β_K*-redex, and* N *has a* β_{eI}*-nf, then* M *has* β_{eI}*-nf.*

Proof: *By induction on the depth of the* F*-redex (following Lemma 2.4).*

- *If* $M \equiv [x_1][x_2]\ldots[x_m](A_1)(A_2)\ldots(A_n)(A)[x]P$, $N \equiv [x_1][x_2]\ldots[x_m](A_1)(A_2)\ldots(A_n)P$ *where* $x \notin FV(P)$, A *is in* β*-nf,* $n \geq 0$ *and* $m \geq 0$. *Use Lemma 5.2 (*A *in* β*-nf* $\Rightarrow$ A *in* β_{eI}*-nf).*

- *Let* $S \equiv [x_1][x_2]\ldots[x_m](A_1)(A_2)\ldots(A_n)$. *If* $M \equiv S(A)[x]P$, $N \equiv S(A')[x]P$ *where* $x \notin FV(P)$, A *is not in* β*-nf,* $A \rightarrow_F A'$, $n \geq 0$ *and* $m \geq 0$. *Use IH to deduce that* A *has* β_{eI}*-nf. As* N *has* β_{eI}*-nf, then* $[x_1][x_2]\ldots[x_m](A_1)(A_2)\ldots(A_n)P$ *has* β_{eI}*-nf by Lemma 5.1. Hence,* $[x_1][x_2]\ldots[x_m](A_1)(A_2)\ldots(A_n)(A)[x]P$ *has* β_{eI}*-nf by Lemma 5.2.*

- *If* $M \equiv [x_1][x_2]\ldots[x_m](A_1)(A_2)\ldots(A_n)(A)(B_1)(B_2)\ldots(B_r)z$,

  $N \equiv [x_1][x_2]\ldots[x_m](A_1)(A_2)\ldots(A_n)(A')(B_1)(B_2)\ldots(B_r)z$ *where*

  A *is not in* β*-nf,* $A \rightarrow_F A'$, $n \geq 0$, $m \geq 0$ *and* $r \geq 0$ *and* $\forall i, 1 \leq i \leq r, B_i$ *is in* β*-nf. As* N *has* β_{eI}*-nf, so does* A' *and by IH, so does* A. *By Lemma 5.3,* M *has* β_{eI}*-nf.* □

Theorem 5.5 *(The inverse of* $\twoheadrightarrow_F$ *preserves* β_{eI}*-N)*
If $M \twoheadrightarrow_F N$ *and* N *is* β_{eI}*-N, then* M *is* β_{eI}*-N.*

Proof: *We show it for* $M \rightarrow_F N$. *Note that if* $M \rightarrow_F N$ *and* $F(M)$ *is a* β_I*-redex, then the theorem is obvious as a* β_I*-redex is a* β_{eI}*-redex. Hence, we only need to prove the theorem for the case when* $F(M)$ *is a* β_K*-redex. But this is already done in Lemma 5.4.* □

Finally, here is the PSN result.

Corollary 5.6 *(Preservation of Strong Normalisation) If* M *is* β*-SN then* M *is* β_e*-SN.*

Proof: *As* M *is* β*-SN, the perpetual strategy of* M *terminates. Let* $M \twoheadrightarrow_F N$ *where* N *is in* β*-nf. As* N *has no* β*-redexes,* N *is* β_{eI}*-N. Hence, by Theorem 5.5,* M *is* β_{eI}*-N. So, by Theorem 4.19* M *is* β_e*-SN.* □

6 Conclusion

In this paper, we established that there is indeed a reduction relation which satisfies both postponement of K-contractions and conservation. This reduction relation is a generalisation of the ordinary β-reduction and has been extensively used since '73 for theoretical and practical reasons (see Section 1.1). We showed moreover that this generalised reduction (called β_e) is indeed a desirable generalisation of β-reduction by showing that β_e preserves strong normalisation in the sense that if M is β-SN then M is β_e-SN. Preservation of Strong Normalisation (PSN) is a property that has to be established for any extension of a reduction relation that is strongly normalising. For example, a lot of research has been carried out lately to establish PSN for β-reduction extended with explicit substitution (see [BLR 95], [KR 95] and [MN 95]). The results of this paper establish that β_e is indeed a safe extension of β.

Finally, it is worth noting that we used item notation in this paper in order to reach the results desired. There is a reason for this. In the usual notation, generalised redexes are not easily visible whereas they are in item notation (see [KN 96b]). For example, in ()()[]()[][], we can clearly see that the leftmost () matches the rightmost[]. Using item notation enables us to write the proofs clearly. Compare with [dG 93] who used a more restricted generalised reduction and still found it hard to discuss where generalised redexes occurs in a term. As a result, de Groote's proofs are longer, more cumbersome and many not included in his paper. In fact, we think that item notation is a good candidate for answering the two questions posed in the conclusions of [Reg 94] concerning the existence of a syntax of terms realising generalised reduction (called σ-reduction by Regnier). It should be noted moreover, that using item notation is not restrictive and that the results of this paper would still hold if we used the classical notation. Only the proofs will be cumbersome to write as the classical notation cannot easily enable us to express generalised redexes.

References

[AFM 95] Ariola, Z.M. Felleisen, M. Maraist, J. Odersky, M. and Wadler, P., A call by need lambda calculus, *Conf. Rec. 22nd Ann. ACM Symp. Princ. Program. Lang. ACM*, 1995.

[Bar 84] Barendregt, H., *Lambda Calculus: its Syntax and Semantics,* North-Holland, 1984.

[Bar 92] Barendregt, H., Lambda calculi with types, *Handbook of Logic in Computer Science*, volume II, ed. Abramsky S., Gabbay D.M., Maibaum T.S.E., Oxford University Press, 1992.

[BLR 95] Benaissa, Briaud, Lescanne, Rouyer-Degli, $\lambda\upsilon$, a calculus of explicit substitutions which preserves strong normalisation, personel communication, 1995.

[BKN 9y] Bloo, R., Kamareddine, F., Nederpelt, R., The Barendregt Cube with Definitions and Generalised Reduction, Computing Science Note, University of Glasgow, Computing Science department, 1994. To appear in *Information and Computation.*

[dG 93] de Groote, P., The conservation theorem revisited, *Int'l Conf. Typed Lambda Calculi and Applications*, vol. 664 of LNCS, 163-178, Springer-Verlag, 1993.

[KN 95] Kamareddine, F., and Nederpelt, R.P., Generalising reduction in the λ-calculus, *Journal of Functional Programming 5 (4)*, 637-651, 1995.

[KN 96b] Kamareddine, F., and Nederpelt, R.P., A useful λ-notation, *Theoretical Computer Science 155*, 1996.

[KR 95] Kamareddine, F., and Rios, A., λ-calculus à la de Bruijn & explicit substitution, *Lecture Notes in Computer Science 982*, 7th international symposium on Programming Languages: Implementations, Logics and Programs, PLILP '95, 45-62, Springer-Verlag, 1995.

[KR 96] Kamareddine, F., and Rios, A., Generalised β_e-reduction and explicit substitution, Computing Science Research Report, University of Glasgow, 1996.

[KTU 94] Kfoury, A.J., Tiuryn, J. and Urzyczyn, P., An analysis of ML typability, J. ACM 41(2), 368-398, 1994.

[KW 94] Kfoury, A.J. and Wells, J.B., A direct algorithm for type inference in the rank-2 fragment of the second order λ-calculus, *Proc. 1994 ACM Conf. LISP Funct. Program.*, 1994.

[KW 95a] Kfoury, A.J. and Wells, J.B., New notions of reductions and non-semantic proofs of β-strong normalisation in typed λ-calculi, *LICS*, 1995.

[KW 95b] Kfoury, A.J. and Wells, J.B., Addendum to new notions of reduction and non-semantic proofs of β-strong normalisation in typed λ-calculi, Boston University.

[MN 95] Muñoz C., Confluence and preservation of strong normalisation in an explicit substitution calculus, Rapport de Recherche No 2762, INRIA.

[Nederpelt 73] Nederpelt, R.P., *Strong normalisation in a typed lambda calculus with lambda structured types*, Ph.D. thesis, Eindhoven University of Technology, Department of Mathematics and Computer Science, 1973. Also appears in [NGV 94].

[NGV 94] Nederpelt, R.P., Geuvers, J.H. and de Vrijer, R.C., eds., *Selected Papers on Automath*, North Holland, 1994.

[Reg 92] Regnier, L., Lambda calcul et réseaux, Thèse de doctorat de l'université Paris 7, 1992.

[Reg 94] Regnier, L., Une équivalence sur les lambda termes, *Theoretical Computer Science 126*, 281-292, 1994.

[SF 92] Sabry, A., and Felleisen, M., Reasoning about programs in continuation-passing style, *Proc. 1992 ACM Conf. LISP Funct. Program.*, 288-298, 1992.

[Vid 89] Vidal, D., *Nouvelles notions de réduction en lambda calcul*, Thèse de doctorat, Université de Nancy 1, 1989.