A LINEARIZATION OF THE LAMBDA-CALCULUS AND CONSEQUENCES

A.J. Kfoury*
Boston University
http://www.cs.bu.edu/~kfoury

February 15, 2000

Abstract

We embed the standard λ-calculus, denoted Λ, into two larger λ-calculi, denoted $\Lambda^\wedge$ and $\&\Lambda^\wedge$. The standard notion of β-reduction for Λ corresponds to two new notions of reduction, $\beta^\wedge$ for $\Lambda^\wedge$ and $\&\beta^\wedge$ for $\&\Lambda^\wedge$. A distinctive feature of our new calculus $\Lambda^\wedge$ (resp., $\&\Lambda^\wedge$) is that, in every function application, an argument is used at most once (resp., exactly once) in the body of the function. We establish various connections between the three notions of reduction, β, $\beta^\wedge$ and $\&\beta^\wedge$. As a consequence, we provide an alternative framework to study the relationship between β-weak normalization and β-strong normalization, and give a new proof of the oft-mentioned equivalence between β-strong normalization of standard λ-terms and typability in a system of "intersection types".

1 Introduction

Background and Motivation

A λ-term M is *linear* (some people say *affine*) if every λ-abstraction in M binds at most one variable occurrence. Linear λ-terms satisfy what we here call the *linearity condition* on function evaluation: If the formal parameter x of an abstraction $(\lambda x.N)$ is not dummy, then the free occurrences of x in the body N of the abstraction are

*Supported by NATO grant CRG 971607, EPSRC grant GR/M08479, NSF grant CCR–9417382, and NSF grant EIA–9806747.

in a one-one correspondence with the arguments to which the function is applied. Many questions about the behavior of linear λ-terms are relatively simple to answer. For example, every linear λ-term is β-strongly normalizing[1] and every closed linear λ-term is simply-typable.[2] Things become more interesting and complicated from the moment we consider λ-abstractions that bind two or more variable occurrences.

Is there a way of simulating the standard λ-calculus by a non-standard λ-calculus where we enforce the linearity condition on function evaluation? What can we gain from this transfer to a non-standard λ-calculus obeying the linearity condition, if at all possible?

Our first goal in Section 2 is therefore to embed the standard λ-calculus Λ in a bigger calculus, denoted $\Lambda^\wedge$, satisfying the linearity condition. Specifically, the way we achieve this is by allowing a subterm P of a λ-term M to be applied to several subterms $Q_1, \ldots, Q_n$ in parallel, which we write as $(P.\ Q_1 \wedge \cdots \wedge Q_n)$. The corresponding notion of β-reduction, denoted $\beta^\wedge$, requires that if P is the λ-abstraction $(\lambda x.N)$ with $m \geqslant 0$ free occurrences of x in N, the reduction cannot be carried out unless $n = \max(m, 1)$. As a consequence, every M in $\Lambda^\wedge$ is $\beta^\wedge$-strongly normalizing. We establish several relationships between β-reduction in Λ and $\beta^\wedge$-reduction in $\Lambda^\wedge$, to determine conditions under which the first can be translated into the second (not always possible) and the second into the first (always possible). An end result is a characterization of β-weak normalization (β-WN) and β-strong normalization (β-SN) for standard λ-terms (Corollary 2.23).

For a finer analysis of the difference between β-WN and β-SN in Section 3, we further embed $\Lambda^\wedge$ in a bigger calculus, denoted $\&\Lambda^\wedge$. In the calculus $\&\Lambda^\wedge$ we deal with expressions of the form $\& M_1 \cdots M_n$ where each of the components $M_1, \ldots, M_n$ is in $\Lambda^\wedge$. The appropriate notion of reduction $\&\beta^\wedge$ is restricted to the leftmost $\beta^\wedge$-redex in $\& M_1 \cdots M_n$, which is moreover adjusted in such a way that arguments of K-redexes are not discarded (Definitions 3.5 and 3.8). Some of the ideas here are suggested by earlier work by several authors, showing how to reduce β-SN to β-WN, but we now adapt them to our special needs. We examine various relationships between β-reduction, $\beta^\wedge$-reduction, and $\&\beta^\wedge$-reduction. A by-product are several

[1] Easy 3-line proof omitted.

[2] A little less straightforward proof, but still easy, also left to the reader.

results connecting the 3 notions of reductions (in particular Theorems 3.6 and 3.16).

Much of the behavior of $\beta^\wedge$-reduction and $\&\beta^\wedge$-reduction is captured by appropriately defined type-inference systems. This is done in Section 4 where we give, among other results, a rigorous proof for the oft-mentioned equivalence between β-SN of standard λ-terms and typability in a system of "intersection types" (Corollary 4.6).

The contribution of this report is more significant for the methodology it develops than for the specific technical results it establishes. What we set up is a new, enlarged framework for the study of β-reduction. There is unavoidably a profusion of new definitions, but once these are understood, the technical results are not surprising and "work as they should".

Future Work

We point out that the present report is unfinished in many ways. Expediency is only partly the reason, as it seems more important in a first report to sketch the broad lines of a new methodology than to examine the implications in detail. We leave some questions unanswered (e.g. Conjecture 2.24), and some results proved only in outline (e.g. Lemma 4.4) or partially proved by methods not promoted in this report (e.g. Corollary 4.6). More important, we do not fully characterize typability in the type-inference systems defined in Section 4 (they do not assign types to all terms) and we leave wide open possible applications of our methodology to other questions (e.g. alternative proofs for the β-SN property of typed λ-calculi).

Related Work

Our *linearity condition* is only one of several kinds of linearity proposed in the literature on λ-calculi in recent years. But one thing they all have in common is the idea of providing "better" resource-management in the evaluation of terms.[3]

Particularly noteworthy is the work of Boudol and his colleagues on the *lambda calculus with multiplicities*. The reader is referred to [3, 4, 5, 6], among other papers dealing with Boudol's approach to linearity. Of these 4 papers, the first 2 stress semantic rather than combinatorial aspects of the proposed calculi; nevertheless, they

[3]We purposely leave the qualification "better" vague, as it may mean any of the following: "simpler to specify", "simpler to analyze", "easier to optimize", "closer to actual implementations", and other similar qualifications.

all shed new light on the mechanism of linearity, not identical but complementary to our analysis in this report.

In addition to Boudol's work, there is the vast literature on *linear logic*. For this kind of linearity, the reader is referred to two of Girard's seminal papers [11, 12] and the references therein. Even though connections between linear logic and our work in this report are not immediately perceived, it will be a useful investigation (left to others) whether results for the former can clarify or be transferred to results for the latter, or vice-versa.

As indicated earlier in this Introduction, our way of dealing with linearity also brings forth connections between β-WN and β-SN. This is so because the notion of reduction $\&\beta^\wedge$ in the expanded calculus $\&\Lambda^\wedge$, presented in full in Section 3, is *non-erasing*, i.e. it does not discard any subexpression or any of its residuals in the course of evaluating an expression. But this is not the only way of preventing "information loss" in the course of an evaluation. In historical order, Nederpelt [23], Klop [19], Karr [13], Khasidashvili [18], de Groote [8], Kfoury and Wells [14, 15], Sørensen [27], and Xi [30], among several others, present different techniques for deducing strong normalization from weak normalization, all inspired by one or both of these two simple ideas:

- *By-pass erasing steps*, i.e. devise a notion of reduction that can be directed to avoid or delay all erasing steps.
- *Memoize erasing steps*, i.e. devise a notion of reduction that will keep a memo of all parts (subterm occurrences) involved in erasing steps. Such a memo or record being part of the syntax of formal expressions, this in effect requires an enlarged calculus where every reduction step becomes non-erasing.

Thus, for example, the approaches proposed in [8] and in [14] are implementations of the first idea, whereas various notions of reductions in [19] and our $\&\beta^\wedge$ in Section 3 of this report can be viewed as implementations of the second. For a fuller account of known results and some of the alternative approaches on this topic, the reader is referred to the highly readable master's thesis by Peter Møller Neergaard [22].

Acknowledgements

Joe Wells played a crucial role in the early stages of the research, by proofreading numerous handwritten drafts and correcting many (sometimes serious) mistakes in them. Although other members of the Church Project will not always recognize the source of the inspiration, many of the ideas in this report are suggested by research they have conducted in recent months and presented in the weekly seminars.[4] Paweł Urzyczyn and two anonymous referees provided many valuable comments and criticisms that shaped the final (present) version of the report.

Some Notational Conventions

- Function $|\ |$ strips all labels from λ-terms (Definition 5.2).
- Function $\mathbf{|\ |}$ contracts expanded λ-terms (Definition 2.3).
- $M \equiv N$ means "M and N are syntactically identical" (up to α-conversion).
- A set of subterm occurrences in M is not a multiset, but a set in the usual sense because different occurrences of the same subterm are distinctly identified. One easy way to think about subterm occurrences is to take M represented by its parse tree (root at the top), with each subterm occurrence in M uniquely identified by its address (a "path") in the parse tree. Just for convenience, addresses of subterm occurrences are left implicit, so that a set of subterm occurrences does appear as a multiset in the text (which it is not).
- If P and Q are subterm occurrences in M, we write $P \subset_M Q$ to mean "P is a proper subterm occurrence of Q in M", i.e. the address of Q in the parse tree of M is a proper prefix of the address of P. We write $P \subseteq_M Q$ for "$P \subset_M Q$ or P is the same occurrence as Q". If M is made clear by the context, or if $M \equiv Q$, we may write $P \subset Q$ and $P \subseteq Q$ instead of $P \subset_M Q$ and $P \subseteq_M Q$, respectively.

[4]More on the Church Project at URL — `http://www.cs.bu.edu/groups/church/`

2 An Expanded λ-Calculus

In our first version $\Lambda^\wedge$ of the expanded λ-calculus, we attempt to supply an abstraction $(\lambda x.N)$ with as many copies of an argument P in order to enforce the linearity condition on function evaluation, by writing an expression of the form $((\lambda x.N).\ P \wedge \cdots \wedge P)$. By structural induction, we can uniformly duplicate arguments of abstractions in this fashion, throughout terms and all their subterm occurrences. However, we cannot determine ahead of time how each copy of an argument P is further used in later evaluation steps, as a function or as an argument or as both. Hence, in order to make the linearity condition invariant of several evaluation steps, we need to write expressions of the form $((\lambda x.N).\ P_1 \wedge \cdots \wedge P_n)$ where $P_1, \ldots, P_n$ are not restricted to be syntactically identical.

The set of λ-variables is λ-Var.

Definition 2.1 (Standard $\boldsymbol{\lambda}$-terms) A standard λ-term M is either a λ-*variable* x or an *abstraction* $(\lambda x.N)$ or an *application* (NP), where $x \in \lambda$-Var and N and P are previously defined standard λ-terms. We denote by Λ the set of standard λ-terms together with the standard rewrite rules on them (β reduction, η reduction, etc.).

Definition 2.2 (Expanded $\boldsymbol{\lambda}$-terms) An expanded λ-term M is either a λ-*variable* x or an *abstraction* $(\lambda x.N)$ or an *expanded application* $(N.P_1 \wedge \cdots \wedge P_n)$, where $x \in \lambda$-Var and N and $P_1, \ldots, P_n$ are previously defined expanded λ-terms, where $n \geqslant 1$. We denote by $\Lambda^\wedge$ the set of expanded λ-terms together with the rewrite rules we later define on them.

We call the subexpression $P_1 \wedge \cdots \wedge P_n$, which is the argument of an expanded application, a $\wedge$-*list* and $P_1, \ldots, P_n$ its *components*. The preceding inductive definition does not include $\wedge$-lists as a 4-th case of expanded λ-terms, but it is easily adjusted so that it does, at the price of making it a bit more complicated. If a $\wedge$-list has only one component, we may write (NP_1) instead of $(N.P_1)$.

Definition 2.3 (Contracting expanded $\boldsymbol{\lambda}$-terms) The *contraction* of an expanded λ-term M is a standard λ-term $\lfloor M \rfloor$, which is defined provided for every subterm of the form $(N.P_1 \wedge \cdots \wedge P_n) \subseteq M$, each of $P_1, \ldots, P_n$ contracts to the same standard term $\lfloor P_1 \rfloor \equiv \cdots \equiv \lfloor P_n \rfloor$. More precisely, by induction on $\Lambda^\wedge$:

1. If $x \in \lambda$-Var, then $|x| = x$.

2. If $x \in \lambda$-Var and $N \in \Lambda^\wedge$, then $|(\lambda x.N)| = (\lambda x.|N|)$ provided $|N|$ is defined, otherwise $|(\lambda x.N)|$ is undefined.

3. If $N, P_1, \ldots, P_n \in \Lambda^\wedge$ and $n \geqslant 1$ then $|(N.P_1 \wedge \cdots \wedge P_n)| = (|N|\ |P_1|)$ provided $|P_1|, \ldots, |P_n|$ are all defined and $|P_1| \equiv \cdots \equiv |P_n|$, otherwise $|(N.P_1 \wedge \cdots \wedge P_n)|$ is undefined.

An expanded λ-term M is *well-formed* if its contraction $|M|$ is defined.

Example 2.4 Let $\mathbf{3} \equiv (\lambda f.\lambda x.f\ (f\ (fx)))$ and $\mathbf{2} \equiv (\lambda g.\lambda y.g\ (gy))$, both of which are standard terms. The following expressions are all in the expanded calculus $\Lambda^\wedge$:

$$M_0 \equiv \mathbf{3}\ \mathbf{2}$$

$$M_1 \equiv \mathbf{3}.\ \mathbf{2} \wedge \mathbf{2} \wedge \mathbf{2}$$

$$M_2 \equiv (\lambda f.\lambda x.(f.\ (f.\ (fx) \wedge (fx)) \wedge (f.\ (fx) \wedge (fx)))).\ \mathbf{2} \wedge \mathbf{2} \wedge \mathbf{2}$$

$$M_3 \equiv (\lambda f.\lambda x.(f.\ (f.\ (fx) \wedge (fx)) \wedge (f.\ (fx) \wedge (fx)))).\ \mathbf{2} \wedge \mathbf{2} \wedge \mathbf{2} \wedge \mathbf{2} \wedge \mathbf{2} \wedge \mathbf{2} \wedge \mathbf{2}$$

$$M_4 \equiv (\lambda f.\lambda x.(f.\ (f.\ (f.x \wedge x) \wedge (f.x \wedge x)) \wedge (f.\ (f.x \wedge x) \wedge (f.x \wedge x)))) \\ .\ \mathbf{2} \wedge \mathbf{2} \wedge \mathbf{2} \wedge \mathbf{2} \wedge \mathbf{2} \wedge \mathbf{2} \wedge \mathbf{2}$$

$$M_5 \equiv (\lambda f.\lambda x.(f.\ (f.\ (fx) \wedge (fx)) \wedge (f\ (fx)))).\ \mathbf{2} \wedge \mathbf{2} \wedge \mathbf{2}$$

$$M_6 \equiv (\lambda f.\lambda x.(f.\ (f.\ (f.x \wedge x) \wedge (fx)) \wedge (f\ (fx)))).\ \mathbf{2} \wedge \mathbf{2} \wedge \mathbf{2} \wedge \mathbf{2} \wedge \mathbf{2} \wedge \mathbf{2}$$

All of the preceding expanded λ-terms are well-formed and all contract to the standard $M \equiv\ \mathbf{3}\ \mathbf{2}$. Let now $\mathbf{1} \equiv (\lambda h.\lambda z.hz)$, which is a standard term. The following are also all in the expanded calculus $\Lambda^\wedge$:

$$N_0 \equiv \mathbf{2}\ \mathbf{1}$$

$$N_1 \equiv \mathbf{2}.\ \mathbf{1} \wedge \mathbf{1} \wedge \mathbf{1}$$

$$N_2 \equiv \mathbf{2}.\ (\mathbf{1}\mathbf{1}) \wedge (\mathbf{1}\mathbf{1}) \wedge (\mathbf{1}\mathbf{1})$$

$$N_3 \equiv \mathbf{2}.\ (\mathbf{1}.\mathbf{1} \wedge \mathbf{1}) \wedge (\mathbf{1}\mathbf{1}) \wedge (\mathbf{1}\mathbf{1})$$

$N_4 \equiv \mathbf{2}.\ (\mathbf{1}\mathbf{1}) \wedge \mathbf{1} \wedge \mathbf{1}$

$N_5 \equiv \mathbf{2}.\ (\mathbf{1}.\mathbf{1} \wedge \mathbf{1}) \wedge \mathbf{1} \wedge \mathbf{1}$

While N_0, N_1, N_2 and N_3 are well-formed, N_4 and N_5 are not, because the contractions $|N_4|$ and $|N_5|$ are not defined. The expanded λ-terms N_0 and N_1 contract to the standard $\mathbf{2}\ \mathbf{1}$, and the expanded λ-terms N_2 and N_3 contract to the standard $\mathbf{2}\ (\mathbf{1}\ \mathbf{1})$.

Definition 2.5 (Parallel sets) Let $M \in \Lambda^\wedge$, not necessarily well-formed. The binary relation $\sim_M$ is the least equivalence on subterm occurrences in M such that:

1. $P_1 \wedge \cdots \wedge P_n \sim_M P_i$ for every $i \in \{1, \ldots, n\}$.
2. If $(\lambda x.N) \sim_M (\lambda x'.N')$ then $N \sim_M N'$.
3. If $(N.\ P_1 \wedge \cdots \wedge P_n) \sim_M (N'.\ P'_1 \wedge \cdots \wedge P'_{n'})$ then $N \sim_M N'$ and $P_1 \wedge \cdots \wedge P_n \sim_M P'_1 \wedge \cdots \wedge P'_{n'}$.

For subterm occurrences N and N' in M, we say N and N' are *parallel occurrences* iff $N \sim_M N'$. A *parallel set* of subterm occurrences in M consists of all the members of a $\sim_M$-equivalence class that are not $\wedge$-lists with 2 or more components.

Remark 2.6 The qualification "that are not $\wedge$-lists with 2 or more components" at the end of Definition 2.5 can be omitted without much conceptual harm but with considerable extra bookkeeping. If we omitted it, we would have to account throughout the paper for the possibility that parallel sets could now mention nested subterm occurrences. For example, in the well-formed N_3 in Example 2.4, $\{\mathbf{1}, \mathbf{1}, \mathbf{1}, \mathbf{1}\}$ is a parallel set, which corresponds to the boxed subterm occurrences in:

$$N_3 \equiv \mathbf{2}.\ (\mathbf{1}.\ \boxed{\mathbf{1}} \wedge \boxed{\mathbf{1}}\ ) \wedge (\mathbf{1}\ \boxed{\mathbf{1}}\ ) \wedge (\mathbf{1}\ \boxed{\mathbf{1}}\ )$$

Without the qualification in question, the same parallel set would be $\{\mathbf{1} \wedge \mathbf{1}, \mathbf{1}, \mathbf{1}, \mathbf{1}, \mathbf{1}\}$ corresponding to

$$N_3 \equiv \mathbf{2}.\ (\mathbf{1}.\ \boxed{\boxed{\mathbf{1}} \wedge \boxed{\mathbf{1}}}\ ) \wedge (\mathbf{1}\ \boxed{\mathbf{1}}\ ) \wedge (\mathbf{1}\ \boxed{\mathbf{1}}\ )$$

Because the parallel set already includes the two components of the $\wedge$-list $\mathbf{1} \wedge \mathbf{1}$, there is no need to include the latter.

Lemma 2.7 Let $M \in \Lambda^\wedge$ be well-formed.

1. There is a one-one correspondence between parallel sets (of subterm occurrences) in M and subterm occurrences in $\lfloor M \rfloor$.

2. If $\mathcal{P} = \{P_1, \ldots, P_n\}$ is a parallel set in M, then $\lfloor P_1 \rfloor \equiv \cdots \equiv \lfloor P_n \rfloor$. It is therefore meaningful to write $\lfloor \mathcal{P} \rfloor$ for the standard λ-term $\lfloor P_1 \rfloor \equiv \cdots \equiv \lfloor P_n \rfloor$.

3. M is standard iff every parallel set in M is a singleton set iff every $\wedge$-list in M has exactly one component.

Proof Part 1 is by induction on M. For part 2, prove that if $P \sim_M P'$ then $\lfloor P \rfloor \equiv \lfloor P' \rfloor$, by induction on the definition of $\sim_M$. Part 3 is immediate from the definitions.

Definition 2.8 (Parallel sets, revisited) It is sometimes easier to use a "bottom-up" inductive definition of parallel sets. We first define a function φ by induction on well-formed $M \in \Lambda^\wedge$ such that $\varphi(M, Q)$ is a set of subterm occurrences in M for every $Q \subseteq \lfloor M \rfloor$:

1. For every $x \in \lambda$-Var, $\varphi(x, x) = \{x\}$.

2. For every well-formed $(\lambda x.N) \in \Lambda^\wedge$ and every $Q \subseteq \lfloor (\lambda x.N) \rfloor$:
$$\varphi((\lambda x.N), Q) = \begin{cases} \varphi(N, Q), & \text{if } Q \subseteq \lfloor N \rfloor, \\ \{(\lambda x.N)\}, & \text{if } Q \equiv \lfloor (\lambda x.N) \rfloor. \end{cases}$$

3. For every well-formed $(N.P_1 \wedge \cdots \wedge P_n) \in \Lambda^\wedge$ and every $Q \subseteq \lfloor (N.P_1 \wedge \cdots \wedge P_n) \rfloor$:
$$\varphi((N.P_1 \wedge \cdots \wedge P_n), Q) = \begin{cases} \varphi(N, Q), & \text{if } Q \subseteq \lfloor N \rfloor, \\ \varphi(P_1, Q) \cup \cdots \cup \varphi(P_n, Q), & \text{if } Q \subseteq \lfloor P_1 \rfloor \equiv \cdots \equiv \lfloor P_n \rfloor, \\ \{(N.P_1 \wedge \cdots \wedge P_n)\}, & \text{if } Q \equiv \lfloor (N.P_1 \wedge \cdots \wedge P_n) \rfloor. \end{cases}$$

A *parallel set* of subterm occurrences in M is $\varphi(M, Q)$ for some $Q \subseteq \lfloor M \rfloor$. The members of the same parallel set are called *parallel occurrences*. We omit the proof that the bottom-up definition here is equivalent to the top-down given in 2.5 when restricted to well-formed expanded λ-terms. (For suggestions on how to formally prove the equivalence of the two definitions, see the section on "Induction and Recursion" in [9] pp. 22-30.) Definition 2.5 does not require that expanded λ-terms be well-formed, while Definition 2.8 does.

Definition 2.9 (Nesting of parallel sets) Let $M \in \Lambda^\wedge$ be well-formed, and $\mathcal{P} = \{P_1, \dots, P_m\}$ and $\mathcal{R} = \{R_1, \dots, R_n\}$ parallel sets in M. We write $\mathcal{P} \prec_M \mathcal{R}$ provided two conditions hold:

1. For every $P \in \mathcal{P}$ there is exactly one $R \in \mathcal{R}$ such that $P \subseteq_M R$.
2. For every $R \in \mathcal{R}$ there is one or more $P \in \mathcal{P}$ such that $P \subseteq_M R$.

The two conditions imply there is an onto map from $\mathcal{P}$ to $\mathcal{R}$, so that also $m \geqslant n$. We write $\mathcal{P} \prec \mathcal{R}$ instead of $\mathcal{P} \prec_M \mathcal{R}$ if the context makes clear $\mathcal{P}$ and $\mathcal{R}$ are parallel sets in M. $\mathcal{P} \preccurlyeq \mathcal{R}$ means "$\mathcal{P} \prec \mathcal{R}$ or $\mathcal{P} = \mathcal{R}$".

Lemma 2.10 Let $M \in \Lambda^\wedge$ be well-formed and $N \equiv \lfloor M \rfloor$.

1. Let P and R be subterm occurrences in M. If $P \subset_M R$ then $\lfloor P \rfloor \subset_N \lfloor R \rfloor$.
2. Let $\mathcal{P}$ and $\mathcal{R}$ be parallel sets in M. Then $\mathcal{P} \prec_M \mathcal{R}$ iff $\lfloor \mathcal{P} \rfloor \subset_N \lfloor \mathcal{R} \rfloor$.

Proof Part 1 is intuitively clear; a formal proof starts with an inductive definition of $\subset_M$, and then proceeds by induction on this definition. For part 2, let $\mathcal{P} = \{P_1, \dots, P_m\}$ and $\mathcal{R} = \{R_1, \dots, R_n\}$. If $\mathcal{P} \prec_M \mathcal{R}$ then $\lfloor P_1 \rfloor \equiv \cdots \equiv \lfloor P_m \rfloor \subset_N \lfloor R_1 \rfloor \equiv \cdots \equiv \lfloor R_n \rfloor$ by part 1 and the definition of $\prec_M$. For the converse, we prove by induction on well-formed $M \in \Lambda^\wedge$ that for arbitrary subterm occurrences P and R in $N \equiv \lfloor M \rfloor$, if $P \subset_N R$ then $\mathcal{P} \prec_M \mathcal{R}$ where $\mathcal{P}$ and $\mathcal{R}$ are the parallel sets in M corresponding to P and R. We use induction on $M \in \Lambda^\wedge$ to produce $\mathcal{P} = \varphi(M, P)$ and $\mathcal{R} = \varphi(M, R)$.

Example 2.11 Consider the expanded λ-term M_6 in Example 2.4. The following are parallel sets of subterm occurrences in M_6:

$\mathcal{P}_1 = \{x, x, x, x\}$	4 occurrences in M_6 contracting to x
$\mathcal{P}_2 = \{(f.\ x \wedge x), (f\ x), (f\ x)\}$	3 occurrences in M_6 contracting to $(f\ x)$
$\mathcal{P}_3 = \{(f.(f.\ x \wedge x) \wedge (f\ x)), (f(f\ x))\}$	2 occurrences in M_6 contracting to $(f(f\ x))$

We have $\mathcal{P}_1 \prec \mathcal{P}_2 \prec \mathcal{P}_3$, corresponding to the fact that $x \equiv \lfloor \mathcal{P}_1 \rfloor$, $(f\ x) \equiv \lfloor \mathcal{P}_2 \rfloor$ and $(f(f\ x)) \equiv \lfloor \mathcal{P}_3 \rfloor$ are subterms of each other in $\lfloor M_6 \rfloor$, i.e. $x \subset (f\ x) \subset (f(f\ x))$.

We identify distinct occurrences of the same variable x in a term M by "occurrence numbers", which are parenthesized positive integers in superscript position, as in

$$M \equiv \ \cdots x^{(1)} \cdots x^{(2)} \cdots \quad \cdots x^{(n)} \cdots$$

Occurrence numbers start with 1, and incremented by 1 as M is scanned from left to right. We α-convert whenever necessary to avoid name ambiguities, which can be achieved by two conditions: (1) every variable name has at most one λ-binding in M, and (2) free variable names are disjoint from bound variable names in M.

Definition 2.12 (Parallel contexts) A *context* C in the expanded calculus is defined as in the standard calculus: C is a term containing some holes. A hole is denoted $\Box$. If the context C has $n \geqslant 1$ holes, we may refer to these holes by $\Box^{(1)}, \ldots, \Box^{(n)}$, numbered in their occurrence order in C from left to right.

Contraction of contexts is defined inductively, as in Definition 2.3, by adding $\lfloor\Box\rfloor = \Box$ to the base case. A context C is *well-formed* if its contraction $\lfloor C \rfloor$ is defined.

A context C with $n \geqslant 1$ holes is a *parallel context* if C is well-formed and $\{\Box^{(1)}, \ldots, \Box^{(n)}\}$ is a parallel set (of subterm occurrences in C).[5]

If C is a context with $n \geqslant 1$ holes and $P_1, \ldots, P_n \in \Lambda^\wedge$ then $C[P_1, \ldots, P_n]$ denotes the result of placing $P_1, P_2, \ldots,$ in $\Box^{(1)}, \Box^{(2)}, \ldots,$ respectively. If the context C is a parallel context and $\lfloor P_1 \rfloor \equiv \cdots \equiv \lfloor P_n \rfloor$ then $C[P_1, \ldots, P_n]$ is well-formed. The converse is not true: $C[P_1, \ldots, P_n]$ may be well-formed even though C is not a well-formed context, let alone a parallel context.

Example 2.13 Let $\mathbf{1}$ and $\mathbf{2}$ be the standard λ-terms defined in Example 2.4. The following is a context C with two holes:

$$C \equiv \mathbf{2}.\ \Box^{(1)} \wedge \Box^{(2)} \wedge \mathbf{1}$$

C is not well-formed because its contraction $\lfloor C \rfloor$ is not defined. C cannot therefore be a parallel context. By placing a copy of $\mathbf{1}$ in each of $\Box^{(1)}$ and $\Box^{(2)}$, we obtain the expanded λ-term $N_1 \equiv \mathbf{2}.\ \mathbf{1} \wedge \mathbf{1} \wedge \mathbf{1}$ in Example 2.4, which is well-formed. Another context C' with 7 holes is:

$$C' \equiv (\lambda f.\lambda x.(f.\ (f.\ \Box^{(1)} \wedge \Box^{(2)}) \wedge (f.\ \Box^{(3)} \wedge \Box^{(4)}))).\ \Box^{(5)} \wedge \Box^{(6)} \wedge \Box^{(7)}$$

[5] An equivalent definition is to say that C is a *parallel context* if C is well-formed and $\lfloor C \rfloor$ is a standard context with exactly one hole.

By placing $(f\ x)$ in the first 4 holes and **2** in the last 3 holes, we obtain the expanded λ-term M_2 in Example 2.4, which is well-formed. However, C' is not a parallel context because $\{\square^{(1)}, \ldots, \square^{(7)}\}$ is not a parallel set (of subterm occurrences in C'), but rather the union of two parallel sets: $\{\square^{(1)}, \square^{(2)}, \square^{(3)}, \square^{(4)}\}$ and $\{\square^{(5)}, \square^{(6)}, \square^{(7)}\}$.

It is a simple result that the set of holes in an arbitrary well-formed context C is always the union of parallel sets of holes in C. We do not need this result later and, therefore, do not include a proof for it here.

Definition 2.14 ($\beta^\wedge$-reduction) We first define a binary relation β' (not yet the desired notion of reduction) on $\Lambda^\wedge$. For arbitrary $N, P_1, \ldots, P_n \in \Lambda^\wedge$, where N mentions $m \geqslant 0$ distinct free occurrences of x, we write:

$$((\lambda x.N).\ P_1 \wedge \cdots \wedge P_n) \underset{\beta'}{\longrightarrow} N[x^{(1)} := P_1, \ldots, x^{(m)} := P_m]$$

provided $n = \max(1, m)$. The notation $N[x^{(1)} := P_1, \ldots, x^{(m)} := P_m]$ refers to the result of substituting P_1 for $x^{(1)}$, P_2 for $x^{(2)}$, etc. We call an expression of the form $((\lambda x.N).\ P_1 \wedge \cdots \wedge P_n)$ a *β'-redex*.[6]

Let $M \equiv C[R_1, \ldots, R_n]$ be a well-formed expanded λ-term, where C is a parallel context with $n \geqslant 1$ holes and $\mathcal{R} = \{R_1, \ldots, R_n\}$ is a parallel set of β'-redex occurrences in M. We write $M \xrightarrow[\beta^\wedge]{\mathcal{R}} N$ provided:

$$N \;\equiv\; C[S_1, \ldots, S_n] \quad \text{and} \quad R_i \underset{\beta'}{\longrightarrow} S_i \quad \text{for every} \quad i \in \{1, \ldots, n\}$$

We call the parallel set $\mathcal{R}$ of β'-redex occurrences a *$\beta^\wedge$-redex occurrence*. If the omission of $\mathcal{R}$ causes no ambiguity, we also write $M \underset{\beta^\wedge}{\longrightarrow} N$. The consistency of this definition is based on Lemma 2.7, according to which $|R_1| \equiv \cdots \equiv |R_n|$, which implies $|S_1| \equiv \cdots \equiv |S_n|$ and, in turn, that $N \equiv C[S_1, \ldots, S_n]$ is well-formed. The transitive reflexive closure of $\underset{\beta^\wedge}{\longrightarrow}$ is $\underset{\beta^\wedge}{\twoheadrightarrow}$.

The difference between $\beta^\wedge$ and β' is that $\beta^\wedge$ requires all β'-redexes in a parallel set to be reduced simultaneously, thus preserving the well-formedness of expanded terms, while β' does not.

[6] Probably it shouldn't be called a "redex", as β' is not a notion of reduction on well-formed expanded λ-terms.

Example 2.15 Consider the expanded λ-terms in Example 2.4. M_0 is already in $\beta^\wedge$-nf. M_1 can be $\beta^\wedge$-reduced to M_1':

$$M_1 \underset{\beta^\wedge}{\longrightarrow} M_1' \equiv \lambda x.\ \mathbf{2}(\mathbf{2}(\mathbf{2}\ x))$$

where M_1' is in $\beta^\wedge$-nf. M_2 is already in $\beta^\wedge$-nf. M_3 can be $\beta^\wedge$-reduced to M_3':

$$\begin{array}{lll} M_3 & \underset{\beta^\wedge}{\longrightarrow} & \lambda x.\ (\mathbf{2}.\ (\mathbf{2}.\ (\mathbf{2}\ x) \wedge (\mathbf{2}\ x)) \wedge (\mathbf{2}.\ (\mathbf{2}\ x) \wedge (\mathbf{2}\ x))) \\ & \underset{\beta^\wedge}{\longrightarrow} & \lambda x.\ (\mathbf{2}.\ R \wedge R) \\ & \underset{\beta^\wedge}{\longrightarrow} & \lambda x.\lambda y.\ R(R\ y) \\ & \underset{\beta^\wedge}{\longrightarrow} & \lambda x.\lambda y.\ \mathbf{2}\ x\ (\mathbf{2}\ x\ (R\ y)) \\ & \underset{\beta^\wedge}{\longrightarrow} & M_3' \equiv \lambda x.\lambda y.\ \mathbf{2}\ x\ (\mathbf{2}\ x\ (\mathbf{2}\ x\ (\mathbf{2}\ x\ y))) \end{array}$$

where $R \equiv \lambda y.\ \mathbf{2}\ x\ (\mathbf{2}\ x\ y)$ and M_3' is in $\beta^\wedge$-nf. M_4 can be $\beta^\wedge$-reduced to M_4' in 4 steps:

$$\begin{array}{lll} M_4 & \underset{\beta^\wedge}{\longrightarrow} & \lambda x.\ (\mathbf{2}.\ (\mathbf{2}.\ (\mathbf{2}.\ x \wedge x) \wedge (\mathbf{2}.\ x \wedge x)) \wedge (\mathbf{2}.\ (\mathbf{2}.\ x \wedge x) \wedge (\mathbf{2}.\ x \wedge x))) \\ & \underset{\beta^\wedge}{\longrightarrow} & \lambda x.\ (\mathbf{2}.\ (\mathbf{2}.\ N \wedge N) \wedge (\mathbf{2}.\ N \wedge N)) \\ & \underset{\beta^\wedge}{\longrightarrow} & \lambda x.\ (\mathbf{2}.\ P \wedge P) \\ & \underset{\beta^\wedge}{\longrightarrow} & M_4' \equiv \lambda x.\lambda y.\ P(P\ y) \end{array}$$

where $N \equiv \lambda y.x(x\ y)$ and $P \equiv \lambda y.N(N\ y)$, and M_4' can be further $\beta^\wedge$-reduced (or also β-reduced) to M_4'' in 6 steps:

$$M_4' \underset{\beta}{\twoheadrightarrow} M_4'' \equiv \lambda x.\lambda y.\ x(x(x(x(x(x(x(x\ y)))))))$$

M_5 is already in $\beta^\wedge$-nf. M_6 is $\beta^\wedge$-reduced to M_6':

$$M_6 \underset{\beta^\wedge}{\longrightarrow} M_6' \equiv \lambda x.\ \mathbf{2}.\ (\mathbf{2}.\ (\mathbf{2}.\ x \wedge x) \wedge (\mathbf{2}\ x)) \wedge (\mathbf{2}\ (\mathbf{2}\ x))$$

where M_6' is in $\beta^\wedge$-nf. Note that M_6' contains a β'-redex, namely $(\mathbf{2}.\ x \wedge x)$, which is also a member of the parallel set $\mathcal{R} = \{(\mathbf{2}.\ x \wedge x), (\mathbf{2}\ x), (\mathbf{2}\ x)\}$. Because not every member of $\mathcal{R}$ is a β'-redex, namely the two occurrences of $(\mathbf{2}\ x)$ are not, $\mathcal{R}$ does not correspond to a $\beta^\wedge$-redex.

The well-formed N_0, N_1 and N_3 are in $\beta^\wedge$-nf. The well-formed N_2 contains a $\beta^\wedge$-redex occurrence, namely, the parallel set $\{(\mathbf{11}), (\mathbf{11}), (\mathbf{11})\}$. As each copy of $(\mathbf{11})$ is β'-reduced (or also β-reduced) to $\mathbf{1}$ in 2 steps, N_2 is $\beta^\wedge$-reduced to N_2' in 2 steps:

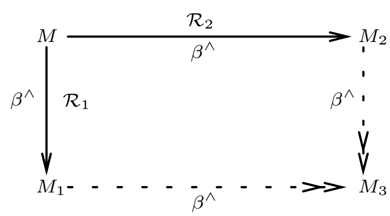
$$N_2 \underset{\beta^\wedge}{\longrightarrow} \underset{\beta^\wedge}{\longrightarrow} N_2' \equiv \mathbf{2}.\ \mathbf{1} \wedge \mathbf{1} \wedge \mathbf{1}$$

where N_2' is in $\beta^\wedge$-nf.

Proposition 2.16 Let M be a well-formed expanded λ-term.

1. M is $\beta^\wedge$-strongly normalizing ("$\beta^\wedge$ is SN").
2. For all M_1 and M_2 such that $M \underset{\beta^\wedge}{\twoheadrightarrow} M_1$ and $M \underset{\beta^\wedge}{\twoheadrightarrow} M_2$, there is M_3 such that $M_1 \underset{\beta^\wedge}{\twoheadrightarrow} M_3$ and $M_2 \underset{\beta^\wedge}{\twoheadrightarrow} M_3$ ("$\beta^\wedge$ is CR").
3. M has exactly one $\beta^\wedge$-nf.

Proof Part 1 follows from the fact that every $\beta^\wedge$-reduction step is strictly size-decreasing. Part 2 implies that M has at most one $\beta^\wedge$-nf and, together with part 1, that M has exactly one $\beta^\wedge$-nf, thus proving part 3. It remains to prove part 2. In fact, by Proposition 3.1.25 in [2], it suffices to show that $\beta^\wedge$ is WCR (weak Church-Rosser), i.e.



where $\mathcal{R}_1$ and $\mathcal{R}_2$ are $\beta^\wedge$-redex occurrences in M:

$$\mathcal{R}_1 = \{\ ((\lambda x.P_1).Q_{1,1} \wedge \cdots \wedge Q_{1,m_1}), \ldots, ((\lambda x.P_n).Q_{n,1} \wedge \cdots \wedge Q_{n,m_n})\ \}$$

$$\mathcal{R}_2 = \{\ ((\lambda x.S_1).T_{1,1} \wedge \cdots \wedge T_{1,p_1}),\ \ldots\ ,((\lambda x.S_q).T_{q,1} \wedge \cdots \wedge T_{q,p_q})\ \}$$

where $n, m_1, \ldots, m_n, q, p_1, \ldots, p_q \geqslant 1$. This is an exhaustive case analysis, generalizing the proof that standard β is WCR, given in Lemma 11.1.1 in [2]. Our proof in fact repeats the proof of Lemma 11.1.1, after the following changes:

— Replace $\Delta_1 \equiv ((\lambda x.P_1)Q_1)$ and $\Delta_2 \equiv ((\lambda x.P_2)Q_2)$ by $\mathcal{R}_1$ and $\mathcal{R}_2$, respectively.

— Replace P_1 and P_2 by $\mathcal{P} = \{P_1, \ldots, P_n\}$ and $\mathcal{S} = \{S_1, \ldots, S_q\}$, respectively.

— Replace Q_1 and Q_2 by $\mathcal{Q} = \{Q_{1,1}, \ldots, Q_{n,m_n}\}$ and $\mathcal{T} = \{T_{1,1}, \ldots, T_{q,p_q}\}$, respectively.

In the various cases and subcases considered in Lemma 11.1.1, replace "$\subset$" by "$\prec$". We omit the straightforward details.

Definition 2.17 (Projecting and lifting) Let t be a $\beta^\wedge$-reduction starting from a well-formed $M_0 \in \Lambda^\wedge$:

$$t: \; M_0 \xrightarrow[\beta^\wedge]{\mathcal{R}_1} M_1 \xrightarrow[\beta^\wedge]{\mathcal{R}_2} M_2 \xrightarrow[\beta^\wedge]{\mathcal{R}_3} \cdots$$

As observed in Definition 2.14, well-formedness of expanded λ-terms is preserved by $\beta^\wedge$ reduction, implying each of $M_1, M_2, \ldots$ is also well-formed. Let u be a β-reduction starting from $N_0 \in \Lambda$:

$$u: \; N_0 \xrightarrow[\beta]{S_1} N_1 \xrightarrow[\beta]{S_2} N_2 \xrightarrow[\beta]{S_3} \cdots$$

We say that u is a *projection* of t, and t a *lifting* of u, if two conditions hold:

1. t and u are sequences with an equal number $k \geqslant 0$ of reduction steps.

2. $\lfloor M_i \rfloor \equiv N_i$ for every $i = 0, 1, \ldots, k$ and $\lfloor \mathcal{R}_i \rfloor \equiv S_i$ for every $i = 1, \ldots, k$.

We say t can be *projected* if there is a projection of t, and u can be *lifted* if there is a lifting of u.

The next proposition makes explicit the simple fact that every $\beta^\wedge$-reduction can be projected. By contrast, as every $\beta^\wedge$-reduction sequence is finite (Proposition 2.16), not every β-reduction can be lifted.

Proposition 2.18 Every $\beta^\wedge$-reduction can be uniquely projected.

Proof Given the $\beta^\wedge$-reduction t as in Definition 2.17, define the sequence $\lfloor t \rfloor$ by:

$$\lfloor t \rfloor: \; \lfloor M_0 \rfloor \xrightarrow[\beta]{\lfloor \mathcal{R}_1 \rfloor} \lfloor M_1 \rfloor \xrightarrow[\beta]{\lfloor \mathcal{R}_2 \rfloor} \lfloor M_2 \rfloor \xrightarrow[\beta]{\lfloor \mathcal{R}_3 \rfloor} \cdots$$

Then $\lfloor t \rfloor$ is well-defined as a β-reduction, i.e. $\lfloor \mathcal{R}_i \rfloor$ is a β-redex occurrence in $\lfloor M_{i-1} \rfloor$ and β-reducing it produces $\lfloor M_i \rfloor$ for every $i = 1, 2, \ldots$. Moreover, every projection of t is obtained from $\lfloor t \rfloor$ by α-renaming.

Definition 2.19 (Expanding $\boldsymbol{\lambda}$-terms) Given a $\wedge$-list $P_1 \wedge \cdots \wedge P_n$, with $n \geqslant 1$, we introduce the shorthand notation $\langle P_1 \wedge \cdots \wedge P_n \rangle_{i,j}$ where $i, j \in \{1, \ldots, n\}$ as an abbreviation for the $\wedge$-list

$$P_1 \wedge \cdots \wedge P_j \wedge P_i \wedge P_{j+1} \wedge \cdots \wedge P_n$$

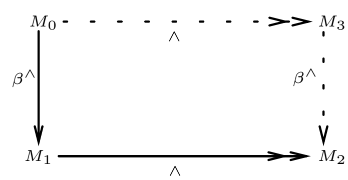
In words, a new copy of the i-th component is inserted right after the j-th component, thus displacing each of the components $P_{j+1}, \ldots, P_n$ one position to the right.

Let M and M' be expanded λ-terms. We write $M \underset{\wedge}{\longrightarrow} M'$ just in case there is a context C with a single hole and expanded λ-terms $S, T_1, \ldots, T_n$, with $n \geqslant 1$, such that

$$M \equiv C[(S.\ T_1 \wedge \cdots \wedge T_n)] \qquad \text{and} \qquad M' \equiv C[(S.\ \langle T_1 \wedge \cdots \wedge T_n \rangle_{i,j})]$$

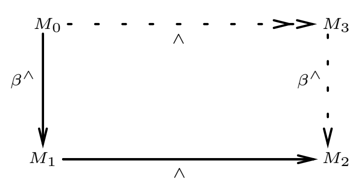
for some $i, j \in \{1, \ldots, n\}$. The context C is not well-formed in general. If we want to name explicitly the application that is expanded, in this case $N \equiv (S.\ T_1 \wedge \cdots \wedge T_n)$, we write $M \overset{N}{\underset{\wedge}{\longrightarrow}} M'$. We use $\underset{\wedge}{\longrightarrow}$ as a notion of "reduction" in the sense of [2] (although it is really an "expansion") and denote its transitive reflexive closure by $\underset{\wedge}{\twoheadrightarrow}$.

Lemma 2.20 For every expanded λ-term M_0 which is well-formed:



In words, we can always displace all expansion steps ahead of $\beta^\wedge$-reduction steps.

Proof It suffices to prove the following simpler commutative diagram:



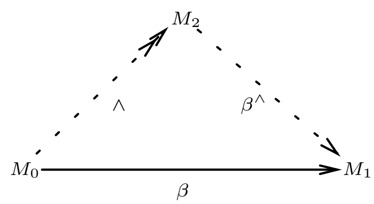
This is a tedious case analysis. Details are in the Appendix of the full report of this paper [16].

In contrast to Lemma 2.20, it is not the case that we can always displace $\beta^\wedge$-reduction steps ahead of expansion steps. Consider for example the sequence:

$$((\lambda x.xxx).\mathbf{I} \wedge \mathbf{I}) \ \underset{\wedge}{\longrightarrow} \ ((\lambda x.xxx).\mathbf{I} \wedge \mathbf{I} \wedge \mathbf{I}) \ \underset{\beta^\wedge}{\longrightarrow} \ \mathbf{III}$$

It is not possible to move the $\beta^\wedge$ step ahead of the expansion step.

Lemma 2.21 For every standard λ-term M_0:



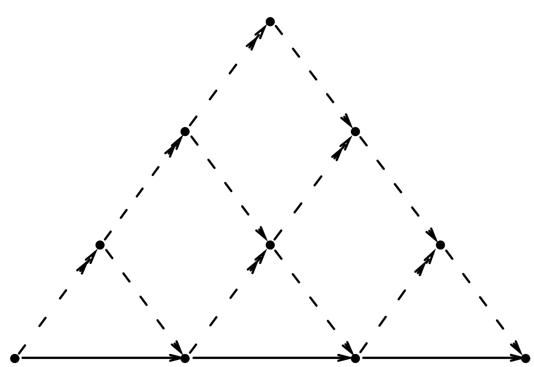
Proof Let $R \equiv ((\lambda x.P)Q)$ be the standard β-redex occurrence in M_0 such that $M_0 \xrightarrow[\beta]{R} M_1$. If R is a K-redex or a I-redex with exactly one free occurrence of x in P, then $\mathcal{R} = \{R\}$ is a $\beta^\wedge$-redex and we just take $M_2 \equiv M_0$. If R is a I-redex, with $n \geqslant 2$ occurrences of the free variable x in P, we expand Q n times to obtain the $\beta^\wedge$-redex

$$\mathcal{R} = \{((\lambda x.P).\ \underbrace{Q \wedge \cdots \wedge Q}_{n}\ )\}$$

In all three cases, $\mathcal{R}$ consists of just one β'-redex. We then carry out the reduction $M_2 \xrightarrow[\beta^\wedge]{\mathcal{R}} M_1$.

Proposition 2.22 Every finite β-reduction can be lifted to a $\beta^\wedge$-reduction (not necessarily unique — see Remark 3.14).

Proof This is a straightforward diagram chase, suggested by the following figure:



The diagram commutes because of Lemma 2.20 (for the parallelograms) and Lemma 2.21 (for the triangles). Each downward arrow is a single $\beta^\wedge$-reduction step, and each two-headed upward arrow is a multiple expansion step. The lower side and the right side of the big triangle are, respectively, the given finite β-reduction and the constructed $\beta^\wedge$-reduction.

A β-reduction u is *maximal* if either u is infinite or u is finite and its last λ-term is in β-nf.

Corollary 2.23 Let M be a standard λ-term.

1. M is β-normalizing iff the maximal leftmost β-reduction starting from M can be lifted.

2. M is β-SN iff every β-reduction starting from M can be lifted.

Proof M is β-normalizing (resp. β-SN) iff the maximal leftmost (resp. every) β-reduction starting from M is finite.

We conjecture a stronger result than part 2 of the preceding corrollary.

Conjecture 2.24 Let M be a standard λ-term. M is β-SN iff there is a well-formed expanded λ-term N such that $M \equiv \lfloor N \rfloor$ and every β-reduction from M can be lifted to a $\beta^{\wedge}$-reduction from N.

The right-to-left implication in 2.24 follows from part 2 of 2.23. The left-to-right implication in 2.24 requires an analysis of the interaction between $\beta^{\wedge}$-reduction and $\wedge$-expansion. We conjecture that the expanded λ-term N constructed in the proof of Theorem 3.16 is a witness for the left-to-right implication in 2.24.

3 A Useful Generalization of Beta-Reduction

K-redexes are the source of many interesting complications in the λ-calculus. The particular complication concerning us here is the difference they introduce between β-weak-normalization (β-WN) and β-strong-normalization (β-SN). In the absence of K-redexes the two notions coincide. There is a long trail of results on how to reduce β-SN to β-WN without excluding K-redexes since the late 1960's, by Nederpelt, by Klop, and by many others in the 1980's and 1990's (see the references already mentioned in the Introduction). We tackle this question once more, not to prove a result (Theorem 3.6) which is likely to be a minor variation of an earlier one in the extensive literature, but to adapt it to our later needs (Theorem 3.16).

Every standard λ-term M which is not in β-nf contains a *leftmost* β-redex occurrence $R \equiv ((\lambda x.P)Q)$. R is uniquely identified by its λ-binding "λx" which occurs to the left of the λ-binding of every other, if any, β-redex occurrence in M.

Lemma 3.1 Let M and N be standard λ-terms, let $R \equiv ((\lambda x.P)Q)$ be a leftmost β-redex occurrence in M, and let $M \xrightarrow[\beta]{R} N$.

1. If R is a I-redex and N is β-SN, then M is β-SN.

2. If R is a K-redex and both N and Q are β-SN, then M is β-SN.

Proof The proof is in the Appendix of the full report of this paper [16].

Example 3.2 Part 2 of the preceding lemma is not true without the restriction "leftmost". Consider the term

$$M \equiv \;\; (\underbrace{(\lambda x.\ (\lambda v.\lambda w.\ vw))\ \mathbf{I}}_{R_1})\ (\lambda y.\ \underbrace{(\lambda x.\mathbf{I})(y\boldsymbol{\omega}\boldsymbol{\omega})}_{R_2})\ (\lambda v.\lambda w.\ vw)$$

where $\mathbf{I} \equiv (\lambda z.z)$ and $\boldsymbol{\omega} \equiv (\lambda z.zz)$. M contains two β-redex occurrences: R_1 and R_2. R_1 is leftmost-outermost, R_2 is only outermost, and both are K-redexes. (A β-redex occurrence R in M is *outermost* if R does not occur as a proper subterm in another β-redex occurrence in M. Leftmost is a special case of outermost.) β-reducing R_2, we get

$$N \equiv \;\; (\ (\lambda x.\ (\lambda v.\lambda w.\ vw))\ \mathbf{I}\ )\ (\lambda y.\ \mathbf{I})\ (\lambda v.\lambda w.\ vw)$$

It is not the case that M is β-SN (it is not) if N and $(y\boldsymbol{\omega}\boldsymbol{\omega})$ are β-SN (they both are). This example also shows that relaxing the "leftmost" restriction to "outermost" is not strong enough to get part 2 of Lemma 3.1.

$G_\beta(M)$ is the β-reduction graph of standard λ-term M (Section 3.1 in [2]). The set of vertices in $G_\beta(M)$ is $\{N \mid M \xrightarrow[\beta]{} \!\!\!\!\!\rightarrow N\}$ modulo α-equivalence. There is an edge from vertex N_1 to vertex N_2 in $G_\beta(M)$ iff $N_1 \xrightarrow[\beta]{} N_2$. $G_\beta(M)$ is a connected graph, because every vertex N is accessible from vertex M. Define

$$\mathit{degree}(M) \;=\; \text{"number of edges in } G_\beta(M)\text{"}$$

The relevant fact for us is: M is β-SN iff $G_\beta(M)$ is a finite dag (directed acyclic graph). In particular, if M is β-SN then $\mathit{degree}(M)$ is finite (the converse is not true).

Lemma 3.3 Let M and N be standard λ-terms, let $R \equiv ((\lambda x.P)Q)$ be a leftmost β-redex occurrence in M, and let $M \xrightarrow[\beta]{R} N$.

1. If R is a I-redex and M is β-SN, then $\mathit{degree}(M) > \mathit{degree}(N)$.

2. If R is a K-redex and M is β-SN, then $\mathit{degree}(M) > \mathit{degree}(N) + \mathit{degree}(Q)$.[7]

Proof The proof is in the Appendix of the full report of this paper [16].

Definition 3.4 (&-lists) We introduce another term constructor & which, by contrast to $\wedge$, can appear only once and in leftmost position in a term. The set of *standard &-lists* is:

$$\&\Lambda \;=\; \{\ \&M_1 \cdots M_n \mid M_1, \ldots, M_n \in \Lambda,\ n \geqslant 1\ \}$$

The set of *expanded &-lists* is:

$$\&\Lambda^\wedge \;=\; \{\ \&M_1 \cdots M_n \mid M_1, \ldots, M_n \in \Lambda^\wedge,\ n \geqslant 1\ \}$$

As $\Lambda \subset \Lambda^\wedge$, we also have $\&\Lambda \subset \&\Lambda^\wedge$. If $M \equiv \&M_1 \cdots M_\ell$, we call the terms $M_1, \ldots, M_\ell$ the *components* of M. A special case is when M has only one component M_1, in which case we may write $M \equiv M_1$ instead of $M \equiv \&M_1$, allowing us to write the following inclusions:

$$\Lambda \subset \&\Lambda \subset \&\Lambda^\wedge \qquad \text{and} \qquad \Lambda \subset \Lambda^\wedge \subset \&\Lambda^\wedge$$

We say that the &-list $M \equiv \&M_1 \cdots M_\ell$ is *well-formed* iff each of $M_1, \ldots, M_\ell$ is well-formed. The *contraction* of $M \equiv \&M_1 \cdots M_\ell$ is simply

$$\lfloor M \rfloor \;\equiv\; \&\ \lfloor M_1 \rfloor\ \cdots\ \lfloor M_\ell \rfloor$$

Clearly, $\lfloor M \rfloor$ is defined iff M is well-formed (review Definition2.3). The definitions of *parallel sets* (Def. 2.5 and Def. 2.8), *parallel contexts* (Def. 2.12), *projecting* and *lifting* (Def. 2.17), and *expanding* (Def. 2.19), are extended to &-lists in the obvious way. Observe that all the members of a parallel set in &-list M are subterm occurrences in the same component of M.

[7] Lemma 3.3 is probably true without the restriction "leftmost" on R, but we do not need such a result.

Definition 3.5 (&β-reduction) Let $M \equiv \& M_1 \cdots M_\ell$ be a standard &-list, and $R \equiv ((\lambda x.P)Q)$ a standard β-redex. We write $M \xrightarrow[\&\beta]{R} M'$ to mean two conditions are satisfied:

1. R is a leftmost β-redex occurrence in M, i.e. there is $k \in \{1, \ldots, \ell\}$ such that R is leftmost in M_k and $M_1, \ldots, M_{k-1}$ are all in β-nf.
2. If $M_k \xrightarrow[\beta]{R} N$, then

$$M' \equiv \begin{cases} \& \; M_1 \; \cdots \; M_{k-1} \, N \, M_{k+1} \; \cdots \; M_\ell, & \text{if } R \text{ is a I-redex,} \\ \& \; M_1 \; \cdots \; M_{k-1} \, N \, Q \; M_{k+1} \; \cdots \; M_\ell, & \text{if } R \text{ is a K-redex.} \end{cases}$$

We write $M \xrightarrow[\&\beta]{} M'$ if there is a leftmost β-redex occurrence R in M such that $M \xrightarrow[\&\beta]{R} M'$.

&β-reduction generalizes β-reduction not only in the sense that (1) it relates two &-lists rather than two λ-terms, but also in the sense that (2) it does not discard arguments of K-redexes after their reduction. On the other hand, only leftmost β-redexes can be &β-reduced, which implies there is a unique &β-reduction starting from a given $M \in \& \Lambda$; in this sense, &β-reduction is more restrictive than β-reduction.

Theorem 3.6 Let M be a standard λ-term. M is β-SN iff M is &β-normalizing.

Proof There are two inductions in this proof, and to push them through, we prove a more general result, namely, for every standard &-list $M \equiv \& M_1 \cdots M_\ell$, the following are equivalent:

(a) Each of the $\ell \geqslant 1$ components $M_1, \ldots, M_\ell$ is β-SN.

(b) M is &β-SN.

(c) M is &β-normalizing.

First, we prove (a) implies (b). Generalize the notion of β-reduction graph to every standard &-list $M \equiv \& M_1 \cdots M_\ell$ by defining

$$G_\beta(M) \;=\; \{\; G_\beta(M_1), \ldots, G_\beta(M_\ell) \;\}$$

Unless M has only one component, $G_\beta(M)$ is a disconnected graph. Define

$$\textit{degree}(M) \;=\; \textit{degree}(M_1) + \cdots + \textit{degree}(M_\ell)$$

It is clear that every component of M is β-SN iff $G_\beta(M)$ is a finite dag.

The proof that (a) implies (b) is by induction on $\textit{degree}(M) \geqslant 0$. If $\textit{degree}(M) = 0$ then every component of M is in β-nf, so that M is also &β-SN. Assume the result true for every standard &-list M where every component is β-SN and $\textit{degree}(M) \leqslant n$. Consider a fixed, but otherwise arbitrary M where every component is β-SN and $\textit{degree}(M) = n + 1$. We want to show that every &β-reduction σ starting from M terminates. Consider the first step of such a reduction σ, say $M \underset{\&\beta}{\longrightarrow} M'$. Reviewing Definition 3.5, it is easy to see that if every component of M is β-SN then so is every component of M' and, by Lemma 3.3, that $\textit{degree}(M') \leqslant n$. Hence, by the induction hypothesis, M' is &β-SN, which in turn implies the reduction σ terminates.

The proof that (b) implies (c) is immediate.

The proof that (c) implies (a) is by induction on the length of &β-normalizing sequences. Consider a &β-normalizing sequence from a standard &-list M:

$$M \equiv P_0 \underset{\&\beta}{\longrightarrow} P_1 \underset{\&\beta}{\longrightarrow} P_2 \underset{\&\beta}{\longrightarrow} \cdots \underset{\&\beta}{\longrightarrow} P_n$$

where P_n is in &β-nf, so that every component of P_n is in β-nf. If $n = 0$, then $P_0 \equiv P_n$ and the desired conclusion is immediate. We assume the result true for every &β-normalizing sequence of length $n \geqslant 0$, and prove it for an an arbitrary &β-normalizing sequence of length $n + 1$, using Lemma 3.1. We omit the obvious details.

Let M be a well-formed expanded λ-term and $\{\mathcal{R}_1, \ldots, \mathcal{R}_n\}$ the set of all $\beta^\wedge$-redex occurrences in M. We say that $\mathcal{R} \in \{\mathcal{R}_1, \ldots, \mathcal{R}_n\}$ is a *leftmost* $\beta^\wedge$*-redex occurrence* in M if $\lfloor\mathcal{R}\rfloor$ is the leftmost among $\{\lfloor\mathcal{R}_1\rfloor, \ldots, \lfloor\mathcal{R}_n\rfloor\}$ in $\lfloor M\rfloor$. Note that $\lfloor M\rfloor$ may contain other β-redex occurrences to the left of $\lfloor\mathcal{R}\rfloor$ which are not the contractions of $\beta^\wedge$-redex occurrences.

Example 3.7 Consider the expanded λ-term N_2 in Example 2.4. There is exactly one $\beta^\wedge$-redex occurrence in N_2, namely, $\mathcal{R} = \{(\mathbf{11}), (\mathbf{11}), (\mathbf{11})\}$. On the other hand, $\lfloor N_2\rfloor \equiv \mathbf{2}(\mathbf{11})$ contains two β-redexes: $\mathbf{2}(\mathbf{11})$ and $\mathbf{11}$. The β-redex $\mathbf{2}(\mathbf{11})$ is to the left of the β-redex $\lfloor\mathcal{R}\rfloor \equiv \mathbf{11}$.

Definition 3.8 (&β^-reduction) Let $M = \& M_1 \cdots M_\ell$ be an expanded &-list, and $\mathcal{R}$ a β^-redex occurrence in M. We write $M \xrightarrow[\&\beta^\wedge]{\mathcal{R}} M'$ to mean two conditions are satisfied:

1. $\mathcal{R}$ is a leftmost β^-redex occurrence in M, i.e. there is $k \in \{1, \ldots, \ell\}$ such that $\mathcal{R}$ is leftmost in M_k and $M_1, \ldots, M_{k-1}$ are all in β^-nf.

2. If $M_k \xrightarrow[\beta^\wedge]{\mathcal{R}} N$, then

$$M' \equiv \begin{cases} \& \; M_1 \; \cdots \; M_{k-1} \, N \, M_{k+1} \; \cdots \; M_\ell, & \text{if } |\mathcal{R}| \text{ is a I-redex,} \\ \& \; M_1 \; \cdots \; M_{k-1} \, N \, Q \, M_{k+1} \; \cdots \; M_\ell, & \text{if } |\mathcal{R}| \text{ is a K-redex and } \mathcal{R} = \{((\lambda x.P)Q)\}. \end{cases}$$

Note, in the case when $|\mathcal{R}|$ is a K-redex, we restrict $\mathcal{R}$ to be a singleton set, i.e. a parallel set consisting of a single β'-redex $((\lambda x.P)Q)$. It is possible to lift this restriction and define instead:

$$M' \;=\; \& \; M_1 \; \cdots \; M_{k-1} \, N \, Q_1 \; \cdots \; Q_n \, M_{k+1} \; \cdots \; M_\ell$$

when $|\mathcal{R}|$ is a K-redex and $\mathcal{R} = \{((\lambda x.P_1)Q_1), \ldots, ((\lambda x.P_n)Q_n)\}$, for arbitrary $n \geqslant 1$, but we do not need this generalization.

We write $M \xrightarrow[\&\beta^\wedge]{} M'$ if there is a leftmost β^-redex occurrence $\mathcal{R}$ in M such that $M \xrightarrow[\&\beta^\wedge]{\mathcal{R}} M'$.

The material to follow, until Theorem 3.16, generalizes material in Section 2. Specifically, Propositions 3.9, 3.11 and 3.15, are generalizations of Propositions 2.16, 2.18 and 2.22. The proofs are very similar, save for a few minor adjustments. Observe that the whole analysis in this section is triggered by the presence of K-redexes: In their absence, 3.9, 3.11 and 3.15 do not say something substantially different from 2.16, 2.18 and 2.22.

Proposition 3.9 Let M be a well-formed expanded &-list.

1. M is &β^-strongly normalizing ("&β^ is SN").

2. M has exactly one &β^-nf.

Proof Part 1 is a consequence of the fact that &β^-reduction is strictly size-decreasing. Part 2 follows from the fact that there is exactly one &β^-reduction starting from M.

A $\&\beta^{\wedge}$-reduction is *lean* if its last &-list is a standard rather than an expanded &-list.

Lemma 3.10 Consider an arbitrary $\&\beta^{\wedge}$-reduction t of length $k \geqslant 1$:

$$t: \ M_0 \xrightarrow[\&\beta^{\wedge}]{\mathcal{R}_1} M_1 \xrightarrow[\&\beta^{\wedge}]{\mathcal{R}_2} M_2 \longrightarrow \cdots \xrightarrow[\&\beta^{\wedge}]{\mathcal{R}_k} M_k$$

If t is lean then, for every $i \in \{1, \ldots, k\}$, $\mathcal{R}_i$ is a parallel set consisting of exactly one β'-redex.

Proof Consider the first $\beta^{\wedge}$-redex in this reduction, say $\mathcal{R}_1$ with no loss of generality, which is not a singleton. Let $\mathcal{R}_1$ be the following parallel set of β'-redex occurrences

$$\mathcal{R}_1 = \ \{ \ ((\lambda x.P_1).Q_{1,1} \wedge \cdots \wedge Q_{1,m_1}), \ldots, ((\lambda x.P_n).Q_{n,1} \wedge \cdots \wedge Q_{n,m_n}) \ \}$$

where $n \geqslant 2$ and $m_1, \ldots, m_n \geqslant 1$. We want to prove that the last &-list M_k in t is not standard, which is equivalent to proving there is a non-singleton parallel set in M_k, because by Lemma 2.7 part 3, an expanded &-list M is standard iff every parallel set in M is a singleton.

We prove therefore there is a parallel set $\mathcal{S}_1$ in M_1, with n members, such that if $\beta^{\wedge}$-redex $\mathcal{R}_2$ is a singleton then $\mathcal{S}_1$ survives to the end of the reduction t, in particular in M_k. If $\beta^{\wedge}$-redex $\mathcal{R}_2$ is not a singleton, we repeat the argument starting from $\mathcal{R}_2$. Consider the set $\mathcal{S}_0$ of subterm occurrences in M_0 defined by:

$$\mathcal{S}_0 = \ \{P_1, \ldots, P_n\} \ = \ \{ \ P_i \mid ((\lambda x.P_i).Q_{i,1} \wedge \cdots \wedge Q_{i,m_i}) \in \mathcal{R}_1 \ \}$$

By Definitions 2.5 and 2.8, it is easy to check that $\mathcal{S}_0$ is a parallel set in M_0. The "residual" of $\mathcal{S}_0$ relative to $M_0 \xrightarrow[\&\beta^{\wedge}]{\mathcal{R}_1} M_1$ is a parallel set $\mathcal{S}_1$ in M_1, given by

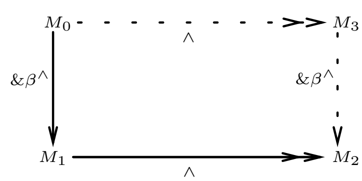
$$\mathcal{S}_1 = \{ \ P_1[x^{(1)} := Q_{1,1}, \ldots, x^{(m_1)} := Q_{1,m_1}], \ldots, P_n[x^{(1)} := Q_{n,1}, \ldots, x^{(m_n)} := Q_{n,m_n}] \ \}$$

There are 3 possible cases: (1) $\mathcal{S}_1 \prec \mathcal{R}_2$, (2) $\mathcal{R}_2 \preccurlyeq \mathcal{S}_1$, (3) neither $\mathcal{S}_1 \prec \mathcal{R}_2$ nor $\mathcal{R}_2 \preccurlyeq \mathcal{S}_1$. In case (3), because $|\mathcal{R}_1|$ is to the left of $|\mathcal{R}_2|$ and $|\mathcal{R}_2|$ is leftmost among $\beta^{\wedge}$-redex occurrences in M_1, $\mathcal{S}_1$ is "untouched" throughout the rest of the reduction t and remains a parallel set in each of $M_2, \ldots, M_k$ — which is the desired conclusion. (A formalization of this argument is in terms of "residuals", as in Definition 5.2, at the cost of making it less transparent.) Case (3) is the only case in which $\mathcal{R}_2$ can be a singleton. In case (2), $\mathcal{R}_2$ has at least $n \geqslant 2$ members (see Definition 2.9), and we repeat the argument starting from $\mathcal{R}_2$. Case (1) cannot happen, because if it did, there would be a $\beta^\wedge$-redex $\mathcal{R}$ in M_0 whose "residual" in M_1 is $\mathcal{R}_2$ and such that $|\mathcal{R}|$ is to the left of $|\mathcal{R}_1|$ in $|M_0|$.

Proposition 3.11 Every lean $\&\beta^\wedge$-reduction can be uniquely projected.

Proof Similar to the proof of Proposition 2.18, using also Lemma 3.10 in order to guarantee that the number of components in each standard &-list (resulting from β-reducing a K-redex) in the projected $\&\beta$-reduction has the same number of components as the corresponding expanded &-list in the given lean $\&\beta^\wedge$-reduction.

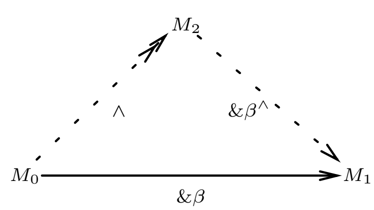
Lemma 3.12 For every well-formed expanded &-list M_0:



where M_3 is uniquely determined.

Proof The proof of Lemma 2.20 can be used here with no change other than replacing "β" by "$\&\beta$" and "$\beta^\wedge$" by "$\&\beta^\wedge$". M_3 is uniquely determined because arguments of K-redexes are not discarded by $\&\beta^\wedge$-reduction.

Lemma 3.13 For every standard &-list M_0:



where M_2 is uniquely determined.

Proof See the proof of Lemma 2.21. The uniqueness of M_2 follows from the fact that arguments of K-redexes are not discarded by $\&\beta$-reduction.

Remark 3.14 In Lemma 2.20 the expanded λ-term M_3 is not uniquely determined, if $\mathcal{R} = \{\ldots, ((\lambda x.P).\ Q_1 \wedge \cdots \wedge Q_n), \ldots\}$ is such that $\lfloor \mathcal{R} \rfloor \equiv ((\lambda x.P')Q')$ is a K-redex, in which case also $n = 1$. See the proof of Lemma 2.20 for the notation here. The reason is that there is no record in M_2 of what expansion is carried out in the argument Q_1 before it disappears. Likewise, in Lemma 2.21, the expanded λ-term M_2 is not uniquely determined. By contrast, the expanded &-lists M_3 in Lemma 3.12 and M_2 in Lemma 3.13 are uniquely determined, because arguments of K-redexes are not lost in $\&\beta$ and $\&\beta^\wedge$ reductions.

Proposition 3.15 Every finite $\&\beta$-reduction can be uniquely lifted to a lean $\&\beta^\wedge$-reduction.

Proof This is a straightforward consequence of Lemmas 3.12 and 3.13. See the proof of Proposition 2.22.

A $\&\beta$-reduction is *maximal* if either it is infinite or it is finite and its last &-list is in $\&\beta$-nf.

Theorem 3.16 Let M be a standard λ-term. M is β-SN iff there is an expanded λ-term N such that $\lfloor N \rfloor \equiv M$ and the maximal $\&\beta$-reduction starting from M can be uniquely lifted to a lean $\&\beta^\wedge$-reduction starting from N.

Proof If M is β-SN then M is $\&\beta$-normalizing, by Theorem 3.6. Hence, by Proposition 3.15, there is a unique $N \in \&\Lambda^\wedge$ such that $M \equiv \lfloor N \rfloor$ and the $\&\beta$-normalizing reduction from M can be uniquely lifted to a lean $\&\beta^\wedge$-reduction from N. Because $M \in \Lambda$ rather than $M \in \&\Lambda$, it must also be that $N \in \Lambda^\wedge$ rather than $N \in \&\Lambda^\wedge$.

Conversely, suppose there is an expanded λ-term N such that $\lfloor N \rfloor \equiv M$ and the maximal $\&\beta$-reduction u from M can be lifted to a $\&\beta^\wedge$-reduction t from N. By Proposition 3.9, t is finite, which implies u is finite. Hence, by Theorem 3.6, M is β-SN.

4 Type Inference Systems

There is a type inference system $\boldsymbol{\lambda}$ based on intersection types for the standard λ-calculus Λ which can be viewed as encoding all reduction sequences of an arbitrary

β-SN λ-term M in the type inferred by $\boldsymbol{\lambda}$ for M. This is reflected in the fact that:

($\star$) A standard λ-term M is β-SN ifff M is typable in $\boldsymbol{\lambda}$.

A discussion of this and related results can be found in [17]. System $\boldsymbol{\lambda}$ is presented below again, along with 3 other type inference systems: $\boldsymbol{\lambda}^{\wedge}$, $\&\boldsymbol{\lambda}$ and $\&\boldsymbol{\lambda}^{\wedge}$, which are special adaptations of $\boldsymbol{\lambda}$ for the expanded λ-calculi $\Lambda^{\wedge}$, $\&\Lambda$ and $\&\Lambda^{\wedge}$, defined in Sections 2 and 3. In a nutshell, these are typing systems powerful enough to encode statically (in the types they derive) the dynamic behavior of all strongly normalizing terms. A by-product of this analysis is to provide differently a new proof of ($\star$) in Corollary 4.6.

Definition 4.1 (Types) We use one type constant, denoted o. We define by simultaneous induction two sets of type expressions, $\mathbb{T}^{\rightarrow}$ and $\mathbb{T}^{\wedge}$:

1. $\mathsf{o} \in \mathbb{T}^{\rightarrow}$.
2. If $\sigma \in \mathbb{T}^{\rightarrow} \cup \mathbb{T}^{\wedge}$ and $\tau \in \mathbb{T}^{\rightarrow}$ then $(\sigma \rightarrow \tau) \in \mathbb{T}^{\rightarrow}$.
3. If $\sigma_1, \ldots, \sigma_n \in \mathbb{T}^{\rightarrow}$ and $n \geqslant 2$ then $(\sigma_1 \wedge \cdots \wedge \sigma_n) \in \mathbb{T}^{\wedge}$.

One more set of type expressions is $\&\mathbb{T}$:

$$\&\mathbb{T} \;=\; \{\ \&\sigma_1 \cdots \sigma_n \mid \sigma_1, \ldots, \sigma_n \in \mathbb{T}^{\rightarrow} \cup \mathbb{T}^{\wedge},\ n \geqslant 2\ \}$$

Let $\mathbb{T} = \mathbb{T}^{\rightarrow} \cup \mathbb{T}^{\wedge} \cup \&\mathbb{T}$. Note that $\mathbb{T}^{\rightarrow} \cup \mathbb{T}^{\wedge}$ is a proper subset of the usual intersection types.

In the various type systems below, A and B denote *type assignments*, i.e. partial functions from λ-Var to $\mathbb{T}$ with finite domain of definition, written as finite lists of pairs. If A and B are type assignments, then $A \wedge B$ is a new type assignment given by:

$$(A \wedge B)(x) = \begin{cases} \text{undefined}, & \text{if both } A(x) \text{ and } B(x) \text{ are undefined,} \\ A(x), & \text{if } A(x) \text{ is defined and } B(x) \text{ is undefined,} \\ B(x), & \text{if } A(x) \text{ is undefined and } B(x) \text{ is defined,} \\ A(x) \wedge B(x), & \text{if both } A(x) \text{ and } B(x) \text{ are defined.} \end{cases}$$

We take $\wedge$ associative, but neither commutative nor idempotent. Hence,

$$(A_1 \wedge A_2) \wedge A_3 \;=\; A_1 \wedge (A_2 \wedge A_3)$$

and we can altogether omit the parentheses. Similarly, if A and B are type assignments, then $\&AB$ is a new type assignment given by:

$$(\&AB)(x) = \begin{cases} \text{undefined}, & \text{if both } A(x) \text{ and } B(x) \text{ are undefined,} \\ A(x), & \text{if } A(x) \text{ is defined and } B(x) \text{ is undefined,} \\ B(x), & \text{if } A(x) \text{ is undefined and } B(x) \text{ is defined,} \\ \&A(x)B(x), & \text{if both } A(x) \text{ and } B(x) \text{ are defined.} \end{cases}$$

To preserve the syntax of types, the operation $(\&AB)$ is used last and at most once on type assignments, e.g., the operation $(\&AB) \wedge (\&A'B')$ is meaningless and there is no type assignment resulting from it.

Our two first systems are $\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}^{\wedge}$. The difference between the two is in the rule APP: In system $\boldsymbol{\lambda}$ a standard application (MN) is assigned a type, in $\boldsymbol{\lambda}^{\wedge}$ an expanded application $(M.\ N_1 \wedge \cdots \wedge N_n)$ is assigned a type. Take note of the side condition $\lfloor N_1 \rfloor \equiv \cdots \equiv \lfloor N_n \rfloor$ in the rule APP of $\boldsymbol{\lambda}^{\wedge}$, which enforces that only *well-formed* expanded λ-terms can be typed.

System $\boldsymbol{\lambda}$

VAR $\quad x : \tau \ \vdash \ x : \tau \qquad \tau \in \mathbb{T}^{\rightarrow}$

ABS-I $$\frac{A,\ x : \sigma_1 \wedge \cdots \wedge \sigma_n \ \vdash \ M : \tau \qquad n \geqslant 1}{A \ \vdash \ (\lambda x.M) : (\sigma_1 \wedge \cdots \wedge \sigma_n \rightarrow \tau)}$$

ABS-K $$\frac{A \ \vdash \ M : \tau \qquad \sigma \in \mathbb{T}^{\rightarrow}}{A \ \vdash \ (\lambda x.M) : (\sigma \rightarrow \tau)}$$

APP $$\frac{A \ \vdash \ M : (\sigma_1 \wedge \cdots \wedge \sigma_n \rightarrow \tau) \qquad B_1 \ \vdash \ N : \sigma_1 \quad \cdots \quad B_n \ \vdash \ N : \sigma_n \qquad n \geqslant 1}{A \wedge B_1 \wedge \cdots \wedge B_n \ \vdash \ (MN) : \tau}$$

System $\boldsymbol{\lambda}^{\wedge}$

VAR	$x : \tau \ \vdash \ x : \tau \qquad \tau \in \mathbb{T}^{\rightarrow}$
ABS-I	$\dfrac{A,\ x : \sigma_1 \wedge \cdots \wedge \sigma_n \ \vdash \ M : \tau \qquad n \geqslant 1}{A \ \vdash \ (\lambda x.M) : (\sigma_1 \wedge \cdots \wedge \sigma_n \rightarrow \tau)}$
ABS-K	$\dfrac{A \ \vdash \ M : \tau \qquad \sigma \in \mathbb{T}^{\rightarrow}}{A \ \vdash \ (\lambda x.M) : (\sigma \rightarrow \tau)}$
APP	$\dfrac{A \ \vdash \ M : (\sigma_1 \wedge \cdots \wedge \sigma_n \rightarrow \tau) \qquad B_1 \ \vdash \ N_1 : \sigma_1 \ \ \cdots \ \ B_n \ \vdash \ N_n : \sigma_n \qquad n \geqslant 1}{A \wedge B_1 \wedge \cdots \wedge B_n \ \vdash \ (M.\ N_1 \wedge \cdots \wedge N_n) : \tau}$
	where $\lfloor N_1 \rfloor \ \equiv \ \cdots \ \equiv \ \lfloor N_n \rfloor$

Systems $\&\boldsymbol{\lambda}$ and $\&\boldsymbol{\lambda}^{\wedge}$ are obtained from $\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}^{\wedge}$, respectively, by adding an additional rule to derive types for &-lists. Rule & is shown in the display below.

&	$\dfrac{A_1 \ \vdash \ M_1 : \sigma_1 \ \ \cdots \ \ A_n \ \vdash \ M_n : \sigma_n \qquad n \geqslant 2}{\&A_1 \cdots A_n \ \vdash \ \&M_1 \cdots M_n : \&\sigma_1 \cdots \sigma_n}$

A distinctive feature of the preceding systems ($\boldsymbol{\lambda}$, $\boldsymbol{\lambda}^{\wedge}$, $\&\boldsymbol{\lambda}$ and $\&\boldsymbol{\lambda}^{\wedge}$) is the following. Suppose there is a derivation $\mathcal{D}$ in any of these 4 systems for the sequent $A \vdash M : \tau$, where x is a λ-variable occurring free in M. (Recall our standing assumption: Free and bound variables are disjoint sets, no variable has more than one λ-binding.) If there are $n \geqslant 1$ invocations of rule VAR in $\mathcal{D}$ to derive n types for x, then $A(x)$ is a type with exactly n "alternatives". For example, if $\mathcal{D}$ is a derivation in $\boldsymbol{\lambda}$ or $\boldsymbol{\lambda}^{\wedge}$, then $A(x)$ is of the form:

$$A(x) = \sigma_1 \wedge \sigma_2 \wedge \cdots \wedge \sigma_n$$

where $\sigma_i \in \mathbb{T}^{\rightarrow}$ for $i = 1, \ldots, n$. Moreover, in the case of $\boldsymbol{\lambda}^{\wedge}$ and $\&\boldsymbol{\lambda}^{\wedge}$, the number of occurrences of x in M is also exactly n. In the case of $\boldsymbol{\lambda}$ and $\&\boldsymbol{\lambda}$, we can only say that $n \geqslant$ the number of occurrences of x in M.

Lemma 4.2

1. Every standard λ-term in β-nf is typable in $\boldsymbol{\lambda}$.

2. Every standard λ-term in β-nf is typable in $\boldsymbol{\lambda}^{\wedge}$.

3. Every standard &-list in &β-nf is typable in &$\boldsymbol{\lambda}$.

4. Every standard &-list in &β-nf is typable in &$\boldsymbol{\lambda}^{\wedge}$.

Proof We define two special subsets of $\mathbb{T}$, $\mathbb{R}$ and $\mathbb{S}$, which are the least such that:

$$\begin{array}{lcl} \mathbb{R} & \supseteq & \{\mathsf{o}\} \cup \{\, (\sigma_1 \wedge \cdots \wedge \sigma_n \to \tau) \mid \sigma_1, \ldots, \sigma_n \in \mathbb{S},\ \tau \in \mathbb{R},\ n \geqslant 1 \,\} \\ \mathbb{S} & \supseteq & \{\mathsf{o}\} \cup \{\, (\sigma \to \tau) \mid \sigma \in \mathbb{R},\ \tau \in \mathbb{S} \,\} \end{array}$$

It is also convenient to work with the following definition of standard λ-terms in β-nf (see Lemma 8.3.18 in [2]).

1. *Passive variable*: If $x \in \lambda$-Var then x is in β-nf.

2. *Abstraction*: If $x \in \lambda$-Var and N is in β-nf then $(\lambda x.N)$ is in β-nf.

3. *Active variable and maximal application*: If $x \in \lambda$-Var and $P_1, \ldots, P_n$ are in β-nf, with $n \geqslant 1$, then $(xP_1 \cdots P_n)$ is in β-nf.

According to clause 3, the first term in a "maximal application" is a variable, which we call "active". The proof of part 1 in by induction on standard λ-terms in β-nf. To push the induction through, we strengthen the induction hypothesis (IH) as follows:

1. The derived type of every:

   (a) passive variable occurrence is o,

   (b) active variable occurrence is in $\mathbb{S}$,

   (c) λ-abstraction is in $\mathbb{R}$,

   (d) maximal application is o.

2. If $A \vdash M : \tau$ is the last sequent in a derivation in $\boldsymbol{\lambda}$, where M is a standard λ-term in β-nf, and there are $n \geqslant 1$ free occurrences of variable x in M, then $A(x) = \sigma_1 \wedge \cdots \wedge \sigma_n$ for some $\sigma_1, \ldots, \sigma_n \in \mathbb{S}$.

We prove for every standard λ-term M in β-nf, there is a derivation in $\boldsymbol{\lambda}$ satisfying IH whose last sequent is $A \vdash M : \tau$, for some A and τ. For the basis of the induction, $M \equiv x$ is a passive variable. In this case, the derivation consisting of the single sequent $x : \mathsf{o} \vdash x : \mathsf{o}$ satisfies IH.

Proceeding inductively, let $M \equiv (\lambda x.N)$. Let $\mathcal{D}$ be a derivation in $\boldsymbol{\lambda}$ satisfying IH whose last sequent is $A \vdash N : \tau$. Hence, in particular, τ is in $\mathbb{R}$. If x does not occur in N, we add the sequent $A \vdash (\lambda x.N) : \mathsf{o} \to \tau$ at the end of $\mathcal{D}$ to obtain a new derivation $\mathcal{D}'$. If x occurs in N, then $A = A_0, x : \sigma_1 \wedge \cdots \wedge \sigma_n$ according to IH, where $\sigma_1, \ldots, \sigma_n \in \mathbb{S}$, and we add the sequent

$$A_0 \;\vdash\; (\lambda x.N) : \; \sigma_1 \wedge \cdots \wedge \sigma_n \to \tau$$

at the end of $\mathcal{D}$ to obtain $\mathcal{D}'$. In either case, $(\mathsf{o} \to \tau)$ or $(\sigma_1 \wedge \cdots \wedge \sigma_n \to \tau)$, the resulting type is in $\mathbb{R}$ and $\mathcal{D}'$ satisfies IH.

Consider next the case when $M \equiv (xP_1 \cdots P_n)$, a maximal application. For $i = 1, \ldots, n$, let $\mathcal{D}_i$ be a derivation in $\boldsymbol{\lambda}$ satisfying IH whose last sequent is $A_i \vdash P_i : \tau_i$. Hence, in particular, τ_i is in $\mathbb{R}$. We construct a derivation $\mathcal{D}$ in $\boldsymbol{\lambda}$ for $(xP_1 \cdots P_n)$ as follows:

$$\cfrac{\cfrac{\{x:\sigma_0\}\vdash x:\sigma_0 \qquad \overset{\mathcal{D}_1}{A_1\vdash P_1:\tau_1}}{\{x:\sigma_0\}\wedge A_1\vdash xP_1:\sigma_1} \qquad \overset{\mathcal{D}_2}{A_2\vdash P_2:\tau_2}}{\{x:\sigma_0\}\wedge A_1\wedge A_2\vdash xP_1P_2:\sigma_2}$$

$$\vdots$$

$$\cfrac{\{x:\sigma_0\}\wedge A_1\wedge\cdots\wedge A_{n-1}\vdash xP_1\cdots P_{n-1}:\sigma_{n-1} \qquad \overset{\mathcal{D}_n}{A_n\vdash P_n:\tau_n}}{\{x:\sigma_0\}\wedge A_1\wedge\cdots\wedge A_n\vdash xP_1\cdots P_n:\sigma_n}$$

where $\sigma_i \equiv \tau_{i+1} \to \cdots \to \tau_n \to \mathsf{o}$ for $i = 0, 1, \ldots, n$. It is easy to check that the new derivation $\mathcal{D}$ satisfies IH: σ_0 is a type in $\mathbb{S}$ because each of $\tau_1, \ldots, \tau_n$ is in $\mathbb{R}$, so that the overall type of x on the left-hand side of $\vdash$ in the last sequent of $\mathcal{D}$ is a $\wedge$-list of types in $\mathbb{S}$. Moreover, the type of the maximal application $(xP_1 \cdots P_n)$ is $\sigma_n \equiv \mathsf{o}$. This concludes the induction and the proof of part 1 of the lemma.

For part 2, first observe that the derivation $\mathcal{D}$ in $\boldsymbol{\lambda}$ which we constructed for an arbitrary standard λ-term M in β-nf has the following property: Every use of the

APP rule in $\mathcal{D}$ has exactly 2 premises. (In general, APP allows 2 or more premises.) Hence, $\mathcal{D}$ is also a valid derivation in $\boldsymbol{\lambda}^{\wedge}$.

Part 3 follows immediately from part 1, and part 4 from part 2.

Lemma 4.3 Let M be an expanded &-list, typable in $\&\boldsymbol{\lambda}^{\wedge}$. If M is in $\&\beta^{\wedge}$-nf then $\lfloor M \rfloor$ is in $\&\beta$-nf.

Proof Suppose $N \equiv \lfloor M \rfloor$ is not in $\&\beta$-nf, i.e. there is a standard β-redex occurrence $R \equiv ((\lambda x.P)Q)$ in N. Let $\mathcal{R}$ be the parallel set in M corresponding to R, $\mathcal{R} = \varphi(M, R)$. If M is typable in $\&\boldsymbol{\lambda}^{\wedge}$, then every

$$((\lambda x.P').\ Q'_1 \wedge \cdots \wedge Q'_n) \ \in \ \mathcal{R}$$

is also typable in $\&\boldsymbol{\lambda}^{\wedge}$. But then the $\wedge$-list $Q'_1 \wedge \cdots \wedge Q'_n$ must have as many components as there are occurrences of x in P', which in turn implies that $\mathcal{R}$ is a $\beta^{\wedge}$-redex occurrence in M and M is not in $\&\beta^{\wedge}$-nf.

Lemma 4.4 Let M and N be expanded &-lists such that $M \xrightarrow[\&\beta^{\wedge}]{} N$. M is typable in $\&\boldsymbol{\lambda}^{\wedge}$ iff N is typable in $\&\boldsymbol{\lambda}^{\wedge}$.

Proof The left-to-right implication ("subject-reduction") is easy to check and therefore omitted. For the inverse implication, suppose N is typable in $\&\boldsymbol{\lambda}^{\wedge}$ and $M \xrightarrow[\&\beta^{\wedge}]{\mathcal{R}} N$ for some $\beta^{\wedge}$-redex $\mathcal{R}$. That M is typable is a straightforward consequence of the linearity condition, described in Section 1, which is satisfied by $\&\beta^{\wedge}$-reduction. (The argument here is identical to the argument showing that for standard λ-terms M and N such that $M \xrightarrow[\beta]{R} N$, where $R \equiv ((\lambda x.P)Q)$ and P mentions exactly one free occurrence of x, M is simply-typable iff N is simply-typable.) A formal proof is by induction on M, which we omit.

An expanded λ-term N is *lean* if the unique $\&\beta^{\wedge}$-reduction from N is lean (see Lemma 3.10 and the definition preceding it).

Theorem 4.5 Le M be a standard λ-term. M is β-SN iff there is a lean expanded λ-term N such that $\lfloor N \rfloor \equiv M$ and N is typable in $\boldsymbol{\lambda}^{\wedge}$.

Proof Suppose M is β-SN. By Theorem 3.16, there is a lean expanded λ-term N such that $|N| \equiv M$ and the maximal $\&\beta$-reduction u from M can be uniquely lifted to a $\&\beta^\wedge$-reduction t from N. The last &-lists in t and u are the same, say N', which is therefore a standard &-list in $\&\beta$-nf. Hence, N' is typable in $\&\boldsymbol{\lambda}^\wedge$, by part 4 in Lemma 4.2. Hence, N is typable in $\&\boldsymbol{\lambda}^\wedge$, by Lemma 4.4 (right-to-left), and because N is not a &-list it is in fact typable in $\boldsymbol{\lambda}^\wedge$.

Conversely, suppose there is a lean expanded λ-term N such that $|N| \equiv M$ and N is typable in $\boldsymbol{\lambda}^\wedge$ — and therefore in $\&\boldsymbol{\lambda}^\wedge$ also. Hence, if N' is the $\&\beta^\wedge$-nf of N, then N' is typable in $\&\boldsymbol{\lambda}^\wedge$, by Lemma 4.4 (left-to-right). Hence, $|N'| \equiv N'$ is also in $\&\beta$-nf, by Lemma 4.3. The maximal $\&\beta^\wedge$-reduction t from N can be uniquely projected to a $\&\beta$-reduction u from M, by Proposition 3.11. N' is the last &-list in both t and u. Hence, as M is $\&\beta$-reduced to the $\&\beta$-nf N', M is $\&\beta$-normalizing and, by Theorem 3.6, M is β-SN.

An oft-mentioned result in the literature (e.g. see [10], [20], [21], [24], [25], [28], [29], and the references cited therein) is that a standard λ-term M is β-SN iff M is typable in (an appropriate formulation of) the system of intersection types. These references in fact provide correct and complete proofs for only one direction of this equivalence, namely, that "if M is typable then M is β-SN". The other direction of this equivalence require a more subtle argument, and the proofs for it in some of these references are buggy and/or incomplete. Some of the bugs are less serious than others; the bug in [20], for example, can be easily remedied with a suitable non-standard notion of length of reduction path.[8]

The first correct proof for this oft-mentioned result in the published literature appears to be the one given in the recent book by Amadio and Curien [1]. The proof of Corollary 4.6 below is another very different one. Strictly speaking, our result is a minor variation of the result mentioned in the literature, as our $\boldsymbol{\lambda}$ is a lean version of the usual system of intersection types, where the type constructor $\wedge$ is also assumed to be commutative and idempotent. It is a simple exercise to transfer our result to

[8]B. Venneri is the one who detected many of the bugs in the earlier proofs for "if M is β-SN then M is typable". She also produced her own correct proof but never published it. I am indebted to Mariangiola Dezani for supplying me with the history reported in this footenote and the paragraph preceding it.

the usual system of intersection types.

Corollary 4.6 Let M be a standard λ-term. M is β-SN iff M is typable in $\boldsymbol{\lambda}$.

Proof We first explain our method for proving the left-to-right implication (the more subtle). For this, we first prove, by induction on $N \in \Lambda^{\wedge}$, that if N is typable in $\boldsymbol{\lambda}^{\wedge}$ then $M \equiv \lfloor N \rfloor$ is typable in $\boldsymbol{\lambda}$. This is a straightforward induction and we omit the details. Observe that N is any well-formed expanded λ-term, not restricted to be lean. Hence, by Theorem 4.5, if M is β-SN then M is typable in $\boldsymbol{\lambda}$.

The right-to-left implication can be proved in various ways. One way is to first prove, by induction, that if M is typable in $\boldsymbol{\lambda}$ then there is a lean expanded λ-term N such that $\lfloor N \rfloor \equiv M$ and N is typable in $\boldsymbol{\lambda}^{\wedge}$, and then to invoke Theorem 4.5 once more (right-to-left). The more expedient way, however, is to use the method of [14] to show that any standard M typable in $\boldsymbol{\lambda}$ is β-SN.[9] Details omitted.

The following corollary is slightly stronger than Theorem 4.5 in that it does not require N to be "lean".

Corollary 4.7 Let M be a standard λ-term. M is β-SN iff there is an expanded λ-term N such that $\lfloor N \rfloor \equiv M$ and N is typable in $\boldsymbol{\lambda}^{\wedge}$.

Proof The left-to-right implication is immediate from Theorem 4.5. For the converse, first use the fact that if there is an expanded λ-term N such that $\lfloor N \rfloor \equiv M$ and N is typable in $\boldsymbol{\lambda}^{\wedge}$, then M is typable in $\boldsymbol{\lambda}$ (see the proof of the preceding corollary). There is no need here to restrict N to be lean. Finally, by Corollary 4.6 (right-to-left), M is β-SN.

5 Appendix: Remaining Proofs

For several of the proofs below we need to define appropriate bookkeeping devices: "nesting-depth" and "residuals".

[9] It is not sufficient to invoke the usual result that "if M is typable in the system of intersection types (without "top") then M is β-SN", because our $\boldsymbol{\lambda}$ is not quite the same as the usual system of intersection types.

Definition 5.1 (Nesting-depth) Let P be a subterm occurrence in standard λ-term M. The nesting-depth of P in M, denoted $\mathit{nesting}(P, M)$ is the number of parenthesis-pairs in M enclosing P, when M is fully parenthesized. A formal definition is by induction on Λ:

1. $\mathit{nesting}(P, x) = \begin{cases} 0, & \text{if } P \equiv x \ , \\ \text{undefined}, & \text{otherwise} \ . \end{cases}$

2. $\mathit{nesting}(P, (MN)) = \begin{cases} 0, & \text{if } P \equiv (MN) \ , \\ 1 + \mathit{nesting}(P, M), & \text{if } P \subset M \ , \\ 1 + \mathit{nesting}(P, N), & \text{if } P \subset N \ , \\ \text{undefined}, & \text{otherwise} \ . \end{cases}$

3. $\mathit{nesting}(P, (\lambda x.M)) = \begin{cases} 0, & \text{if } P \equiv (\lambda x.M) \ , \\ 1 + \mathit{nesting}(P, M), & \text{if } P \subset M \ , \\ \text{undefined}, & \text{otherwise} \ . \end{cases}$

Definition 5.2 (Residuals) The approach in Chapter 11 of [2] for keeping track of a β-redex occurrence, as the term of which it is a subterm is repeatedly β-reduced, is to label its leading "λ". For our purposes, we need to keep track of other subterm occurrences, not only β-redex occurrences, in λ-terms that are β-reduced (or β'-reduced) as well as expanded. For a uniform labelling scheme here, we choose to keep track of a subterm by placing a label under it (if it is a variable) or under its enclosing parentheses (if it is not a variable), as in

$$\underset{i}{x} \qquad\qquad \underset{i}{(}\ \lambda x.\ N\ \underset{i}{)} \qquad\qquad \underset{i}{(}\ N.\ P_1 \wedge \cdots \wedge P_n\ \underset{i}{)}$$

where $i \in \mathbb{N}$, the set of natural numbers. Formally, by induction on $\&\Lambda^\wedge$:

1. If $x \in \lambda\text{-Var}$ then $\underset{[i]}{x} \in \overline{\Lambda^\wedge}$.

2. If $x \in \lambda\text{-Var}$, $N \in \overline{\Lambda^\wedge}$ and $i \in \mathbb{N}$ then $\underset{[i]}{(}\ \lambda x.\ N\ \underset{[i]}{)} \in \overline{\Lambda^\wedge}$.

3. If $N, P_1, \ldots, P_n \in \overline{\Lambda^\wedge}$ and $i \in \mathbb{N}$ then $\underset{[i]}{(}\ N.\ P_1 \wedge \cdots \wedge P_n\ \underset{[i]}{)} \in \overline{\Lambda^\wedge}$.

4. If $M_1, \ldots, M_\ell \in \overline{\Lambda^\wedge}$ then $\&M_1 \cdots M_\ell \in \overline{\&\Lambda^\wedge}$.

The notation $[i]$ means the label i may or may not be present, but if it is present in one occurrence of $[i]$ it is present in the other. $\overline{\&\Lambda^\wedge}$ is $\&\Lambda^\wedge$ after labels are introduced.

A λ-term can be both a β-redex (or β'-redex) and an application. We choose to identify it as a β-redex by the label on the parentheses enclosing its abstraction, and as an application by the label on its outermost enclosing parentheses. For example, the β-redex $((\lambda x.N)P)$ can be given two label-pairs, as in

$$\underset{2}{(}\,\underset{1}{(}\,\lambda x.\,N\,\underset{1}{)}\,P\,\underset{2}{)}$$

"1" identifies $((\lambda x.N)P)$ as a β-redex, "2" identifies it as an application. If it is β-reduced, both label-pairs are lost:

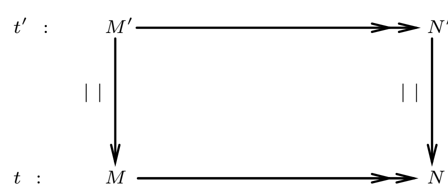
$$\underset{2}{(}\,\underset{1}{(}\,\lambda x.\,N\,\underset{1}{)}\,P\,\underset{2}{)} \xrightarrow[\beta]{} N[x := P]$$

If it is expanded, both label-pairs are preserved:

$$\underset{2}{(}\,\underset{1}{(}\,\lambda x.\,N\,\underset{1}{)}\,P\,\underset{2}{)} \xrightarrow[\wedge]{} \underset{2}{(}\,\underset{1}{(}\,\lambda x.\,N\,\underset{1}{)}.\,P \wedge P\,\underset{2}{)}$$

As in [2], if $M \in \overline{\&\Lambda^{\wedge}}$ we denote $|M|$ the expression obtained by erasing all labels in M.

Consider a mixed sequence t of $\beta^{\wedge}$-reduction (or $\&\beta^{\wedge}$-reduction) steps and expansion steps from $M \in \Lambda^{\wedge}$ to $N \in \Lambda^{\wedge}$ (or from $M \in \&\Lambda^{\wedge}$ to $N \in \&\Lambda^{\wedge}$). Let R and S be subterm occurrences in M and N, respectively. We say that S is a *residual* of R relative to t if there is a mixed sequence t' from $M' \in \overline{\Lambda^{\wedge}}$ to $N' \in \overline{\Lambda^{\wedge}}$ (or from $M' \in \overline{\&\Lambda^{\wedge}}$ to $N' \in \overline{\&\Lambda^{\wedge}}$) such that



where R is the only labelled subterm occurrence in M', with some $i \in \mathbb{N}$, and S is one of the labelled subterm occurrences in N', with the same i.

If $\mathcal{R}$ and $\mathcal{S}$ are parallel sets of subterm occurrences in M and N, respectively, we say that $\mathcal{S}$ is a *residual* of $\mathcal{R}$ relative to t if for every $S \in \mathcal{S}$ there is $R \in \mathcal{R}$ such that S is a residual of R relative to t.

Proof of Lemma 2.20: Let $\mathcal{R}$ be the $\beta^{\wedge}$-redex occurrence in M_0 such that $M_0 \xrightarrow[\beta^{\wedge}]{\mathcal{R}} M_1$. Recall that $\mathcal{R}$ is a parallel set of β'-redex occurrences in M_0, and therefore if

$R \in \mathcal{R}$ then R is of the form $((\lambda x.P).\ Q_1 \wedge \cdots \wedge Q_n)$ such that $\lvert R\rvert \equiv \lvert \mathcal{R}\rvert \equiv ((\lambda x.P')Q')$ for some standard $P' \equiv \lvert P\rvert$ and $Q' \equiv \lvert Q_1\rvert \equiv \cdots \equiv \lvert Q_n\rvert$.

Let N_1 be the (expanded) application in M_1 such that $M_1 \xrightarrow[\wedge]{N_1} M_2$. It is easy to see there is a uniquely defined application $N_0 \subset M_0$ such that N_1 is the residual of N_0 relative to the reduction $M_0 \xrightarrow[\beta\wedge]{\mathcal{R}} M_1$ and, moreover, N_1 is the only residual of N_0 relative to this reduction. Let $N_0 \equiv (S.\ T_1 \wedge \cdots \wedge T_k)$.

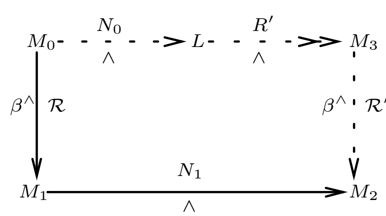
A β'-redex occurrence $R \in \mathcal{R}$ is also an application occurrence in M_0, but because it is reduced, R has no residual in M_1. Hence, $N_0 \notin \mathcal{R}$ and the only possible cases to consider are:

1. $N_0 \subset R$ for some $R \in \mathcal{R}$.

2. $R \subset N_0$ for some $R \in \mathcal{R}$.

3. Neither $N_0 \subset R$ nor $R \subset N_0$ for every $R \in \mathcal{R}$.

Consider the first case when $N_0 \subset R \equiv ((\lambda x.P).\ Q_1 \wedge \cdots \wedge Q_n)$ for some $R \in \mathcal{R}$. There are two subcases here: $N_0 \subset P$ or $N_0 \subset Q$ for some $Q \in \{Q_1, \ldots, Q_n\}$. If $N_0 \subset P$ *and* one of $T_1, \ldots, T_k$ (and therefore all of them) contains free occurrences of x, then



where R' is the residual of R relative to $M_0 \xrightarrow[\wedge]{N_0} L$, and $\mathcal{R}'$ is the residual of $\mathcal{R}$ relative to $M_0 \xrightarrow[\wedge]{N_0} L \overset{R'}{\underset{\wedge}{\twoheadrightarrow}} M_3$. Because $N_0 \subset P$, the expansion $M_0 \xrightarrow[\wedge]{N_0} L$ will increase the number of free occurrences of x in P, which in turn requires that the number of components in the $\wedge$-list $Q_1 \wedge \cdots \wedge Q_n$ be increased accordingly — this explains why the expansion $L \overset{R'}{\underset{\wedge}{\twoheadrightarrow}} M_3$ is necessary before we can carry out the reduction $M_3 \xrightarrow[\beta\wedge]{\mathcal{R}'} M_2$. We omit the details as to which components in $Q_1 \wedge \cdots \wedge Q_n$ have to be duplicated.

For all remaining subcases and cases:

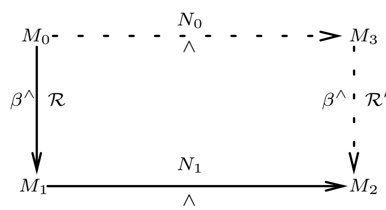
(a) $N_0 \subset P$ *and* none of $T_1, \ldots, T_k$ contains a free occurrence of x,

(b) $N_0 \subset Q$ for some $Q \in \{Q_1, \ldots, Q_n\}$,

(c) $R \subset N_0$ for some $R \in \mathcal{R}$ (case 2 above),

(d) Neither $N_0 \subset R$ nor $R \subset N_0$ for every $R \in \mathcal{R}$ (case 3 above),

it is easy to check that the following commutative diagram obtains



where $\mathcal{R}'$ is the residual of $\mathcal{R}$ relative to $M_0 \xrightarrow[\wedge]{N_0} M_3$. Each of (a), (b), (c) and (d), has to be checked separately. We omit the straightforward details. Note that in case (c) (case 2), there may be more than one $R \in \mathcal{R}$ which is a subterm occurrence in N_0.

Proof of Lemma 3.1: Part 1 of this lemma is immediate from the Conservation Theorem (Theorem 13.4.12 in [2]). The restriction "leftmost" is not necessary for part 1.

Prove part 2 by induction on $\textit{nesting}(R, M) \geqslant 0$. The base case is $\textit{nesting}(R, M) = 0$, which means $R \equiv M$, for which the result is easy to check.

Suppose part 2 of the lemma is true for every $M \in \Lambda$ such that $\textit{nesting}(R, M) \leqslant k$, for some $k \geqslant 0$ — this is the induction hypothesis (IH). Consider next a fixed, but otherwise arbitrary, $M \in \Lambda$ such that $\textit{nesting}(R, M) = k + 1$. If $M \equiv (\lambda y.M_0)$, then $\textit{nesting}(R, M_0) = k$, and the desired result follows from the IH.

Consider the case when $M \equiv (M_0 M_1)$. Either $\textit{nesting}(R, M_0) = k$ or $\textit{nesting}(R, M_1) = k$. Because R is the leftmost β-redex occurrence in M and $\textit{nesting}(R, M) \neq 0$, it follows that $M \not\equiv R$ and M_0 is not a λ-abstraction. M is therefore of the form (parentheses omitted for clarity):

$$M \;\equiv\; L_0 L_1 \cdots L_\ell L_{\ell+1}$$

where L_0 is either a variable or a β-redex, $M_0 \equiv (\cdots(L_0 L_1)\cdots L_\ell)$ and $M_1 \equiv L_{\ell+1}$, where $\ell \geqslant 0$. If L_0 is a variable and $\mathit{nesting}(R, M_0) = k$ or $\mathit{nesting}(R, M_1) = k$, then in fact $\mathit{nesting}(R, L_i) \leqslant k$ for some $i \in \{1, \ldots, \ell+1\}$ and the desired conclusion follows from the IH.

The remaining case is when L_0 is a β-redex. Because R is leftmost, in fact $L_0 \equiv R$. It is now easy to see that if both $N \equiv PL_1 \cdots L_\ell M_1$ and Q are β-SN then so is M.

Proof of Lemma 3.3:[10] We refer to a graph G by writing $G = (V, E)$, where V and E are respectively its set of vertices and its set of edges. $G' = (V', E')$ is a *subgraph* of $G = (V, E)$ if $V' \subseteq V$ and $E' \subseteq E$. Given a set $V' \subseteq V$, the subgraph of G *induced* by V' is the graph $G' = (V', E')$ where $E' = E \cap (V \times V')$.

Part 1 of this lemma is immediate from the fact that $G_\beta(M)$ is a finite dag. The rest of this proof concerns part 2 only. We prove a stronger result: If $G_\beta(M) = (V_M, E_M)$, $G_\beta(N) = (V_N, E_N)$ and $G_\beta(Q) = (V_Q, E_Q)$, then there are proper subsets $V_a, V_b \subset V_M$ such that

(a) $G_\beta(N)$ is isomorphic to the subgraph of $G_\beta(M)$ induced by V_a,

(b) $G_\beta(Q)$ is isomorphic to the subgraph of $G_\beta(M)$ induced by V_b, and

(c) $V_a \cap V_b = \varnothing$.

The condition in (c) guarantees that the two subgraphs in (a) and (b) do not have edges in common. This, together with the fact that $G_\beta(M)$ is a connected graph, implies that $\mathit{degree}(M) > \mathit{degree}(N) + \mathit{degree}(Q)$.

[10] I tried to include most of the important details in this proof. As it is right now, it is quite long, in any case longer than what I wished. It would be nice to have a shorter proof, especially that the main idea is quite simple.

The proof of (a), (b) and (c), is by induction on $nesting(R, M) \geqslant 0$. The base case is $nesting(R, M) = 0$, for which $M \equiv R \equiv ((\lambda x.P)Q)$ and $N \equiv P$. In this case

$$
\begin{aligned}
V_M &= V_P \cup \{ ((\lambda x.P')Q') \mid P' \in V_P,\ Q' \in V_Q \} \\
E_M &= E_P \cup \\
&\quad \{ ((\lambda x.P')Q') \to ((\lambda x.P'')Q') \mid P' \to P'' \in E_P \} \cup \\
&\quad \{ ((\lambda x.P')Q') \to ((\lambda x.P')Q'') \mid Q' \to Q'' \in E_Q \} \cup \\
&\quad \{ ((\lambda x.P')Q') \to P' \mid P' \in V_P,\ Q' \in V_Q \}
\end{aligned}
$$

It is clear that $G_\beta(N) = G_\beta(P) = (V_P, E_P)$ is a proper subgraph of $G_\beta(M)$, and that $G_\beta(Q)$ is isomorphic to the subgraph of $G_\beta(M)$ (it is not the only one) induced by the set of vertices $\{((\lambda x.P)Q') | Q' \in V_Q\} \subset V_M$. It remains to show that $V_P \cap \{((\lambda x.P)Q') | Q' \in V_Q\} = \varnothing$.

If M is β-SN there cannot be $P' \in V_P$ and $Q' \in V_Q$ such that $P' \equiv ((\lambda x.P)Q')$, otherwise we would have the following infinite β-reduction from M:

$$
\begin{aligned}
M \ \equiv\ (\lambda x.P)Q \ &\underset{\beta}{\twoheadrightarrow}\ (\lambda x.P)Q' \\
&\underset{\beta}{\twoheadrightarrow}\ (\lambda x.P')Q' \equiv (\lambda x.(\lambda x.P)Q')Q' \\
&\underset{\beta}{\twoheadrightarrow}\ (\lambda x.(\lambda x.P')Q')Q' \equiv (\lambda x.(\lambda x.(\lambda x.P)Q')Q')Q' \\
&\ \ \vdots \\
&\underset{\beta}{\twoheadrightarrow}\ \underbrace{(\lambda x.(\lambda x.\cdots(\lambda x.(\lambda x.}_{n\geqslant 1} P)\underbrace{Q')\cdots Q')Q'}_{n\geqslant 1} \\
&\ \ \vdots
\end{aligned}
$$

Hence, $V_P \cap \{((\lambda x.P)Q') | Q' \in V_Q\} = \varnothing$, as desired. This concludes the proof of the base case.

Suppose the result true for every $M \in \Lambda$ such that $nesting(R, M) \leqslant k$, for some $k \geqslant 0$ — this is the induction hypothesis (IH). Consider next a fixed, but otherwise arbitrary, $M \in \Lambda$ such that $nesting(R, M) = k + 1$. If $M \equiv (\lambda y.M_0)$, then $nesting(R, M_0) = k$, and it is easy to check that the desired result follows from the IH. (No need to fill in the details here, as the more complicated cases below make it clear how to do it.)

If $M \equiv (M_0 M_1)$ then either $nesting(R, M_0) = k$ or $nesting(R, M_1) = k$. Because R is leftmost in M and $nesting(R, M) \neq 0$, we have that $M \not\equiv R$ and M_0 is not a λ-abstraction. M is therefore of the form:

$$M \;\equiv\; L_0 L_1 \cdots L_\ell L_{\ell+1}$$

where L_0 is either a variable or a β-redex, $M_0 \equiv (\cdots(L_0 L_1)\cdots L_\ell)$ and $M_1 \equiv L_{\ell+1}$, where $\ell \geqslant 0$.

If L_0 is a variable y and $nesting(R, M_0) = k$ or $nesting(R, M_1) = k$, then $nesting(R, L_i) \leqslant k$ for some $i \in \{1, \ldots, \ell+1\}$. With no loss of generality, let R be a β-redex occurrence in L_1 and $L_1 \xrightarrow[\beta]{R} \widetilde{L_1}$, so that

$$M \equiv y L_1 L_2 \cdots L_\ell L_{\ell+1} \;\xrightarrow[\beta]{R}\; N \equiv y \widetilde{L_1} L_2 \cdots L_\ell L_{\ell+1}$$

The vertex set and edge set of $G_\beta(M)$ are:

$$\begin{aligned} V_M &= \{ y L'_1 \cdots L'_{\ell+1} \mid L'_i \in V_{L_i},\; i = 1, 2, \ldots, \ell+1 \} \;\supset V_N \\ E_M &= \{ (y L'_1 \cdots L'_{\ell+1}) \to (y L''_1 \cdots L''_{\ell+1}) \mid (\exists i)[L'_i \to L''_i \in E_{L_i} \;\&\; (\forall j \neq i)[L'_j = L''_j]] \} \;\supset E_N \end{aligned}$$

It is immediate that $G_\beta(N)$ is a proper subgraph of $G_\beta(M)$. By the IH, $G_\beta(\widetilde{L_1})$ and $G_\beta(Q)$ are isomorphic to proper subgraphs of $G_\beta(L_1)$ induced by disjoint sets of vertices, say $V_a, V_b \subset V_{L_1}$. Hence $G_\beta(N) = G_\beta(y\widetilde{L_1} L_2 \cdots L_{\ell+1})$ and $G_\beta(Q)$ are isomorphic to proper subgraphs of $G_\beta(M) = G_\beta(y L_1 L_2 \cdots L_{\ell+1})$ induced by the following sets of vertices:

$$\widehat{V_a} \;=\; \{\; y L'_1 L_2 \cdots L_{\ell+1} \mid L'_1 \in V_a \;\} \qquad \text{and} \qquad \widehat{V_b} \;=\; \{\; y L'_1 L_2 \cdots L_{\ell+1} \mid L'_1 \in V_b \;\}$$

Because $V_a \cap V_b = \varnothing$, we also have $\widehat{V_a} \cap \widehat{V_b} = \varnothing$.

The remaining case is when L_0 is a β-redex. Because R is leftmost, $L_0 \equiv R$ and therefore $M \equiv ((\lambda x.P)Q)L_1 \cdots L_{\ell+1}$ and $N \equiv PL_1 \cdots L_{\ell+1}$. For notational convenience, let $L_{-2} \equiv P$ and $L_{-1} \equiv Q$. The vertex and edge sets of $G_\beta(M)$ are:

$$\begin{aligned}
V_M &= \{ L'_{-2}L'_1 \cdots L'_{\ell+1} \mid L'_i \in V_{L_i},\ i = -2, 1, 2, \ldots, \ell+1 \} \ \cup \\
&\quad\ \{ (\lambda x.L'_{-2})L'_{-1}L'_1 \cdots L'_{\ell+1} \mid L'_i \in V_{L_i},\ i = -2, -1, 1, 2, \ldots, \ell+1 \} \\
E_M &= \{ L'_{-2}L'_1 \cdots L'_{\ell+1} \to L''_{-2}L''_1 \cdots L''_{\ell+1} \mid \\
&\qquad\qquad (\exists i)[L'_i \to L''_i \in E_{L_i} \ \&\ (\forall j \neq i)[L'_j = L''_j]] \} \ \cup \\
&\quad\ \{ (\lambda x.L'_{-2})L'_{-1}L'_1 \cdots L'_{\ell+1} \to (\lambda x.L''_{-2})L''_{-1}L''_1 \cdots L''_{\ell+1} \mid \\
&\qquad\qquad (\exists i)[L'_i \to L''_i \in E_{L_i} \ \&\ (\forall j \neq i)[L'_j = L''_j]] \} \ \cup \\
&\quad\ \{ (\lambda x.L'_{-2})L'_{-1}L'_1 \cdots L'_{\ell+1} \to (L'_{-2}L'_1 \cdots L'_{\ell+1}) \mid L'_i \in V_{L_i},\ i = -2, -1, 1, 2, \ldots, \ell+1 \}
\end{aligned}$$

The first set on the righthand side of the first equation is precisely V_N, and the first set on the righthand side of the second equation is precisely E_N. Hence, $G_\beta(N)$ is a subgraph of $G_\beta(M)$. The IH, together with the fact that $M_0 \xrightarrow[\beta]{R} L_{-2}L_1 \cdots L_\ell$, imply that $G_\beta(L_{-2}L_1 \cdots L_\ell)$ and $G_\beta(Q)$ are isomorphic to proper subgraphs of

$$G_\beta(M_0) \ = \ G_\beta((\lambda x.L_{-2})L_{-1}L_1 \cdots L_\ell)$$

induced by disjoint subsets of vertices, say $V_a, V_b \subset V_{M_0}$. Hence, $G_\beta(N) = G_\beta(L_{-2}L_1 \cdots L_\ell L_{\ell+1})$ and $G_\beta(Q)$ are isomorphic to proper subgraphs of $G_\beta(M)$ induced by the sets of vertices:

$$\widehat{V_a} = V_a \cdot \{L_{\ell+1}\} \qquad \text{and} \qquad \widehat{V_b} = V_b \cdot \{L_{\ell+1}\}$$

Because $V_a \cap V_b = \varnothing$, it follows that $\widehat{V_a} \cap \widehat{V_b} = \varnothing$.

References

[1] Amadio, R. M., and Curien, P.-L., *Domains and Lambda-Calculi*, Cambridge University Press, 1998.

[2] Barendregt, H.P., *The Lambda Calculus, Its Syntax and Semantics*, revised edition, North-Holland, Amsterdam, 1984.

[3] Boudol, G., Curien, P.-L., and Lavatelli, C., "A semantics for Lambda Calculi with Resources", *Mathematical Structures in Comput. Sci. (MSCS)*, Vol. 9 (1999), pp 437-482.

[4] Boudol, G., and Lavatelli, C., "The Discriminating Power of Multiplicities in the Lambda-Calculus", *Information and Computation*, Vol. 126, No. 1 (1996), pp 83-102.

[5] Boudol, G., "The Pi-Calculus in Direct Style", *Proceedings of 24th ACM Symposium on Principles of Programming Languages (POPL)*, January 1997, pp 228-241.

[6] Boudol, G., "Typing the Use of Resources in a Concurrent Calculus", *Asian Computing Science Conference (ASIAN'97)*, Kathmandu, Nepal, LNCS 1345 (1997), pp 239-253.

[7] Coppo, M. and Giannini, P., "A Complete Type Inference Algorithm for Simple Intersection Types", *17th Colloquium on Trees in Algebra and Programming, CAAP 92*, LNCS 581, pp 102-123, 1992.

[8] de Groote, Ph., "The Conservation Theorem Revisited", *Typed Lambda Calculi and Applications, TLCA '93*, eds. Bezem and Groote, LNCS 664, pp 163-178, Springer Verlag 1993.

[9] Enderton, H.B., *A Mathematical Introduction to Logic*, Academic Press, 1972.

[10] Ghilezan, S., "Strong Normalization and Typability with Intersection Types", *Notre Dame J. Formal Logic*, Vol 37, no. 1, Winter 1996.

[11] Girard, J.-Y., "Linear Logic", *Theoretical Computer Science*, Vol 50, no. 1, pp 1-102, 1987.

[12] Girard, J.-Y., "Linear Logic: Its Syntax and Semantics", in *Advances in Linear Logic*, eds. J.-Y. Girard, Y. Lafont and L. Regnier, Cambridge University Press, pp 1-42, 1993.

[13] Karr, M., "Delayability in Proofs of Strong Normalizability in the Typed Lambda Calculus", *Mathematical Foundations of Computer Software*, eds. Ehrig, Floyd, Nivat and Thatcher, LNCS 185, pp 208-222, Springer Verlag 1985.

[14] Kfoury, A.J. and Wells, J.B., "New Notions of Reduction and Non-Semantic Proofs of Beta Strong Normalization in Typed Lambda Calculi", Proceedings of *Logic in Computer Science*, June 1995.

[15] Kfoury, A.J. and Wells, J.B., Addendum to "New Notions of Reduction and Non-Semantic Proofs of Beta Strong Normalization in Typed Lambda Calculi", BUCS Tech Report 95-007, Computer Science Department, Boston University, March 1995.

[16] Kfoury, A.J., "A Linearization of the Lambda-Calculus and Consequneces", BUCS Tech Report 96-007, Computer Science Department, Boston University, March 1995.

[17] Kfoury, A.J., "Beta-Reduction as Unification", in *Logic, Algebra, and Computer Science (H. Rasiowa Memorial Conference), December 1996*, Banach Center Publication, Volume 46, ed. D. Niwinski, Springer Verlag, 1999.

[18] Khasidashvili, Z., *Form Reduction Systems and Reductions of Contracted Forms and Lambda-Terms*, PhD thesis, Tbilisi State University, 1991 (in Russian).

[19] Klop, J. W., *Combinatory Reduction Systems*, PhD thesis, Utrecht University, Amsterdam 1980. Volume 127 of CWI Tract.

[20] Krivine, J.-L., *Lambda-calcul, Types et Modèles*, Masson, Paris, 1990.

[21] Leivant, D., "Typing and Computational Properties of Lambda Expressions", *Theoretical Computer Science*, Vol 44, pp 51-68, 1986.

[22] Møller Neerdgaard, P., *Weak and Strong Normalization,* **K***-redexes, and First-Order Logic*, Master's thesis, DIKU, August 1999, DIKU Technical Report 99/11, available at:
`http://people.bu.edu/turtle/academia.html#master-thesis`
`ftp://ftp.diku.dk/diku/semantics/papers/D-399.ps.gz`

[23] Nederpelt, R., *Strong Normalization in a Typed Lambda Calculus With Lambda Structured Types*, PhD thesis, Eindhoven 1973.

[24] Pottinger, G., "A Type Assignment for the Strongly Normalizable λ-terms", in *To H.B. Curry: Essays on Combinatory Logic, Lambda Calculus, and Formalism*, J.P. Seldin and J.R. Hindley, eds., pp 561-577, Academic Press, 1980.

[25] Retoré, C., "A Note on Intersection Types", INRIA report RR-2431, January 1995, postscript available at:
`ftp://ftp.inria.fr/INRIA/publication/publi-ps-gz/RR/RR-2431.ps.gz`

[26] Ronchi della Rocca, S., "Principal Type Schemes and Unification for Intersection Type Discipline", *Theoretical Computer Science*, Vol. 59, pp 181-209, 1988.

[27] Sørensen, M.H., "Strong Normalization from Weak Normalization in Typed Lambda Calculi", *Information and Computation*, Vol. 133, no. 1, pp 35-71, February 1997.

[28] van Bakel, S., "Complete Restrictions of the Intersection Type Discipline", *Theo. Comp. Sc.*, Vol 102, pp 135-163, 1992.

[29] van Bakel, S., *Intersection Type Disciplines in Lambda Calculus and Applicative Term Rewriting Systems*, Doctoral dissertation, Catholic University of Nijmegen, also issued by the Mathematisch Centrum, Amsterdam, 1993.

[30] Xi, Hongwei, "Weak and Strong Beta Normalisations in Typed λ-calculi", *Third Int'l Conference on Typed Lambda Calculi and Applications, TLCA '97*, LNCS 1210, Springer Verlag 1997.