Math. Struct. in Comp. Science (1993), *vol.* 11, *pp.* 1–000

Simulating expansions without expansions

Roberto Di Cosmo†

Delia Kesner ¶

Received

We add extensional equalities for the functional and product types to the typed λ-calculus with not only products and terminal object, but also sums and bounded recursion (a version of recursion that does not allow recursive calls of infinite length). We provide a confluent and strongly normalizing (thus decidable) rewriting system for the calculus, that stays confluent when allowing unbounded recursion. For that, we turn the extensional equalities into *expansion* rules, and not into contractions as is done traditionally. We first prove the calculus to be weakly confluent, which is a more complex and interesting task than for the usual λ-calculus. Then we provide an effective mechanism to simulate expansions without expansion rules, so that the strong normalization of the calculus can be derived from that of the underlying, traditional, non extensional system. These results give us the confluence of the full calculus, but we also show how to deduce confluence directly form our simulation technique, without the weak confluence property.

1. Introduction

Over the past years there has been a growing interest in the properties of λ-calculus extended with various different type constructors, in particular pairs and sums, used to represent common data types. For these type constructors it is customary to provide a set of equalities that are then turned into computation rules: this is the case, for example, of the elimination rules for pairs:

$$(\pi_1) \quad \pi_1(\langle M, N\rangle) \quad = \quad M \qquad (\pi_2) \quad \pi_2(\langle M, N\rangle) \quad = \quad N$$

They tell us how to operationally *compute* with objects of these types: if we have a pair $\langle M, N\rangle$, then we can decompose it to access its first or second component.

There is anyway something else that one likes to do with λ-calculus, besides using λ-terms as programs to be computed: one would like to *reason* about programs, to prove that they enjoy certain properties. Here is where extensional equalities come into play. In the case of functions, for example, since the only operational way to *use* a function is to apply it to an argument, we do not really want to consider a term M of function type different from the term $\lambda x.Mx$ where x does not occur free in M: both terms, when applied to an argument

† DMI-LIENS (CNRS URA 1347) Ecole Normale Supérieure - 45, Rue d'Ulm - 75230 Paris France
E-mail:`dicosmo@dmi.ens.fr`
¶ INRIA Rocquencourt - Domaine de Voluceau, BP 105 - 78153 Le Chesnay Cedex, France and
CNRS and LRI - Bât 490, Université de Paris-Sud - 91405 Orsay Cedex, France
E-mail:`kesner@lri.lri.fr`

N, give the same result MN. Similarly for pairs, the only operational way to *use* a pair is by projecting out the first or the second component, so we do not want to consider a term M of product type different from the term $\langle \pi_1(M), \pi_2(M) \rangle$: the result of accessing any of these two terms via a first or second projection is the same term $\pi_1(M)$ or $\pi_2(M)$.

These facts can be incorporated in the calculus in the form of equalities, that one can read in at least two different ways:

— *an operational way*: these equalities just state possible *optimizations* of a program. Since a term $\langle \pi_1(M), \pi_2(M) \rangle$ is more complex then M, but behaves the same way, it is convenient to replace all its occurrences by M, as this transformation will yield an equivalent, but more efficient and smaller program. Similarly, we will replace every occurrence of $\lambda x.Mx$ by M.

— *a theoretical way*: these equalities state a relation between a program and its type. They just tell us that whenever a term M has a functional type, then it must really be a function, built by λ-abstraction, so we ought to replace it by $\lambda x.Mx$ if it is not already a function. Similarly, a term M of product type has to be really a pair, built via the pair constructor, or otherwise it must be replaced by $\langle \pi_1(M), \pi_2(M) \rangle$.

As we will briefly see in the Survey, a lot of research activity has focused on the operational reading of these equalities in the tradition of λ-calculus, while only a little on the theoretical one. In this paper we will show how this last reading of the equalities provides a confluent and strongly normalizing reduction system for the simply typed λ-calculus with pairs, sums, unit type (or terminal object) and a bounded recursion operator. We also show that the same reduction system stays confluent when allowing unbounded recursion, while of course loosing the strong normalization property.

1.1. *Survey*

Due to the deep connections between λ-calculus, proof theory and category theory, works on extensional equalities have appeared with different motivations in all these fields.

By far, the best known extensional equality is the η axiom that we informally introduced above, written in the λ-calculus formalism as

$$(\eta) \qquad \lambda x.Mx = M \qquad \text{provided } x \text{ is not free in } M$$

This axiom, also known as *extensionality*, has traditionally been turned into a reduction, carrying the same name, by orienting the equality from left to right, interpreting operationally equality as a *contraction*. Such an interpretation is well behaved as it preserves confluence (CF58).

In the early 70's, the attention was focusing on products and the extensional rule for pairs, called *surjective pairing*, which is the analog for product types of the usual η extensional rule.

$$(SP) \qquad \langle \pi_1(M), \pi_2(M) \rangle = M$$

With the previous experience of the η rule, it is easy to understand how, at that time, most of the people thought that the right way to turn such an equality into a rewrite rule was also from left to right, as a contraction. But in 1980, J.W. Klop discovered (Klo80) that, if added to the usual confluent rewrite rules for pure λ-calculus, this interpretation of *SP* breaks confluence†.

Anyway, this first negative result was shortly after mitigated in (Pot81) for the simply *typed* λ-calculus with

† See (Bar84), p. 403-409 for a short history and references.

η and SP contractions, by providing a first proof of confluence and strong normalization, later on simplified in different ways (see (Tro86) or (GLT90), for example). From then on, the contraction rule for SP was not considered harmful in a typed framework, until the seminal work by Lambek and Scott (LS86). There, the decision problem of the equational theory of Cartesian Closed Categories (ccc's) is solved using a particular typed λ-calculus equipped with not only η and SP equalities, but also with a special type $\mathbf{T}$ representing the *terminal object* of the ccc's[‡]. This distinguished atomic type comes with a further extensional axiom asserting that there is exactly one term $*$ of type **T**:

$$(Top) \qquad M : \mathbf{T} = *$$

Now, the type $\mathbf{T}$ has the bad property of destroying confluence, if the extensional equalities η and SP are turned into contraction rules: the following are the critical pairs that arise immediately, as first pointed out by Obtulowicz, (see (LS86)):

$$\begin{array}{ll}
\langle \pi_1(x), \pi_2(x) \rangle \Rightarrow_{SP} x & \qquad \langle \pi_1(x), \pi_2(x) \rangle \Rightarrow_{SP} x \\
\Downarrow^{Top} & \qquad \Downarrow^{Top} \\
\langle *, \pi_2(x) \rangle & \qquad \langle \pi_1(x), * \rangle
\end{array}$$

$$\begin{array}{ll}
(\lambda x : \mathbf{T}.Mx) : \mathbf{T} \to A \Rightarrow_\eta M & \qquad (\lambda x : A.Mx) : A \to \mathbf{T} \Rightarrow_\eta M \\
\Downarrow_{Top} & \qquad \Downarrow_{Top} \\
(\lambda x : \mathbf{T}.M*) : \mathbf{T} \to A & \qquad (\lambda x : A.*) : A \to \mathbf{T}
\end{array}$$

It is indeed possible, but not easy, to extend the contractive reduction system in order to recover confluence. A first step towards such a confluent system was taken by Poigné and Voss, who were not inspired by category theory, but by the implementation of algebraic data types (PV87). In their paper, they study a calculus that includes $\lambda^1\beta\eta\pi*$, and notice that to solve the previous critical pairs one needs to add an infinite number of reduction rules (that can be anyway finitely described). Then confluence of such an extended system can be proved by showing weak confluence and strong normalization. Unfortunately, the critical pair for $(\lambda x : A.Mx) : \mathbf{T} \to A$ is missing there, and the strong normalization proof is incomplete.

More recently, Curien and the first author got interested in a polymorphic extension of $\lambda^1\beta\eta\pi*$, that arose in the study of the theory of object oriented programming and of isomorphisms of types (CDC91). They give a complete (infinite) set of reduction rules for the calculus, which is proved confluent using just weak confluence, weak normalization and some additional properties.

Meanwhile, in the field of proof theory, Prawitz was suggesting (Pra71) to turn these extensional equalities into *expansion* rules, rather than contractions. Building on such ideas, but motivated by the study of coherence problems in category theory, Mints gives a first faulty proof that in the typed framework *expansion rules*, if handled with care, are weakly normalizing and preserve confluence of the typed calculus (Min79)[§].

This idea of using expansion rules seems to have passed unnoticed for a long time, even if the so called η-long normal forms were well known and used in the study of higher order unification problems (Hue76): only in

‡ This is the *Unit* type in languages like ML.

§ The same idea is present in (Min77).

these last years there has been a renewed interest in expansion rules. In recent work (Jay92), still motivated by category theoretic investigation, Jay explores a simply typed λ-calculus with just **T** and a natural number type **N** as base types, equipped with an induction combinator for terms of type **N**. He introduced expansion rules for η and SP that are exactly the same as the ones originally used by Mints, and in (JG92) this calculus is proved confluent and strongly normalizing. Category theory is also the motivation of Cubric (Cub92), who repaired the error in the original proof by Mints, showing confluence (by means of a careful study of residuals) and also weak normalization (but not strong normalization). An interesting divide-and-conquer approach is proposed in (Aka93), where confluence and strong normalization are shown in a modular way. Finally, in (Dou93), confluence is shown via the usual Newman's Lemma and strong normalization by means of a variation of the reducibility proof based on introduction rather than elimination terms.

1.2. *Our work*

The present paper is inspired especially by (Jay92) and (PV87). We use expansion rules to provide a confluent rewriting system for the typed λ-calculus with not only products and terminal object, but also sums and recursion. This result is derived from the confluence of a restricted system where recursion is bounded (recursive calls of infinite length are not allowed), which is proved to be weakly confluent and strongly normalizing.

We show that strong normalization of the full system can be reduced to that of the system without expansion rules, for which the traditional techniques can be used. For that purpose, we show that any one step reduction in the calculus with expansions can be *simulated* by a non-empty reduction sequence in the calculus without expansions. It turns out that this result is powerful enough to prove directly also the confluence property, as shown in section 7.

Since the reduction with expansion rules is not a congruence, several fundamental properties that hold for the well known typed λ-calculi have to be reformulated in the expansionary framework in a different way as we will see in Section 4. For this reason we believe that the system with expansion rules deserves to be studied much more carefully, so we will undertake the task of proving directly weak confluence: this will lead us to uncover many of the essential features of this reduction.

We introduce now the calculus and its reduction system in section 2, then we investigate the key properties of the new reduction system: weak confluence (section 4) and strong normalization (section 5). In section 7 we derive the confluence property in two different ways and finally in the conclusion we discuss some further applications of our proof techniques.

An extended abstract of this work can be found in (DCK93a).

2. The Calculus

It is now time to introduce the calculus we will deal with in this paper. There are two versions, one with bounded recursion, and the other with unbounded recursion, that differ just in the term formation rule and in the equality rule for recursive terms. We will now introduce the calculus with bounded recursion and then describe how the unbounded version can be obtained from it.

2.1. *Types and Terms*

The set of types of our calculus contains a distinguished type constant $\mathbf{T}$[¶], a denumerable set of atomic or base types, and is closed w.r.t. formation of function, product and sum, i.e. if A and B are types, then also $A \rightarrow B$, $A \times B$ and $A + B$ are types.

For each type A, we fix a denumerable set of variables of that type. We will use $x, y, z, \ldots$ to range over variables, and for a term M we write $M : A$ to mean that M is a term of type A.

The term formation rules of the calculus can then be presented as follows.

$$\frac{}{\Gamma \vdash * : \mathbf{T}}$$

$$\frac{}{x_1 : A_1, \ldots, x_n : A_n \vdash x_i : A_i} \quad 1 \leq i \leq n \quad \text{and the } x_i\text{'s are pairwise distinct}$$

$$\frac{\Gamma, x : A \vdash M : B}{\Gamma \vdash \lambda x : A.M : A \rightarrow B} \qquad \frac{\Gamma \vdash M : A \rightarrow B \quad \Gamma \vdash N : A}{\Gamma \vdash (MN) : B}$$

$$\frac{\Gamma \vdash M : A \quad \Gamma \vdash N : B}{\Gamma \vdash \langle M, N \rangle : A \times B} \qquad \frac{\Gamma \vdash M : B_1 \times B_2}{\Gamma \vdash \pi_i(M) : B_i} \quad i = 1, 2$$

$$\frac{\Gamma \vdash M : B_i}{\Gamma \vdash in^i_{B_1+B_2}(M) : B_1 + B_2} \quad i = 1, 2 \qquad \frac{\Gamma \vdash P : A_1 + A_2 \quad \Gamma \vdash M_i : A_i \rightarrow D}{\Gamma \vdash Case(P, M_1, M_2) : D}$$

$$\frac{\Gamma, x : A \vdash M : A}{\Gamma \vdash (rec\ x : A.M)^i : A} \quad i \geq 0$$

We may omit types of variables in λ-abstractions when they are clear from the context writing $\lambda y.M$ instead of $\lambda y : C.M$.

Notation 2.1. (Free variables, substitutions)
The set of free variables of a term M will be noted $FV(M)$. It can be defined inductively as follows:

[¶] This stands for the terminal object in ccc's or for the *Unit* type in languages like ML.

$$
\begin{array}{ll}
FV(*) & = \emptyset \\
FV(x) & = \{x\} \\
FV(O_A) & = FV(M) \\
FV(MN) & = FV(M) + FV(N) \\
FV(\langle M, N \rangle) & = FV(M) + FV(N) \\
FV(\lambda x : A.M) & = FV(M) - \{x\} \\
FV((rec\ x : A.M)^i) & = FV(M) - \{x\} \\
FV(in^1_C(M)) & = FV(M) \\
FV(in^2_C(M)) & = FV(M) \\
FV(\pi_1(M)) & = FV(M) \\
FV(\pi_2(M)) & = FV(M) \\
FV(Case(P, M, N)) & = FV(P) + FV(M) + FV(N)
\end{array}
$$

We write $[N_1, \ldots, N_n/x_1, \ldots, x_n]$ (often abbreviated $[\overline{N}/\overline{x}]$) for the typed substitution mapping each variable $x_i : A_i$ to a term $N_i : A_i$. We write $M[\overline{N}/\overline{x}]$ for the term M where each variable x_i free in M is replaced by N_i.

2.2. *Equality*

Besides the usual identification of terms up to α conversion (i.e. renaming of bound variables), our calculus is equipped with the equality E on terms generated from the following axioms.

$$
\begin{array}{llcl}
(\beta) & (\lambda x : A.M)N & = & M[N/x] \\
(\pi_1) & \pi_1(\langle M_1, M_2 \rangle) & = & M_1 \\
(\pi_2) & \pi_2(\langle M_1, M_2 \rangle) & = & M_2 \\
(\rho) & Case(in^1_C(R), M_1, M_2) & = & M_1 R \\
 & Case(in^2_C(R), M_1, M_2) & = & M_2 R \\
(rec) & (rec\ y : C.M)^{i+1} & = & M[(rec\ y : C.M)^i/y] \\
(\eta) & \lambda x : A.Mx & = & M \text{ if } \left\{ \begin{array}{l} x \notin FV(M) \\ M : A \to B \end{array} \right. \\
(\delta) & \langle \pi_1(M), \pi_2(M) \rangle & = & M \text{ if } M : A \times B \\
(Top) & M & = & * \text{ if } M : \mathbf{T}
\end{array}
$$

The index i that is attached to each rec term is a *bound* on the depth of the recursive calls that can originate from it. With such a bound, it is possible to insure the strong normalization of the associated reduction system.

The unbounded system is obtained from the bounded one by simply erasing all the bound indexes from the formation and equality rules (and the associated reduction rules). As we will show later, the bounded system can simulate any finite reduction of the unbounded system, and this fact will make it easy to extend the confluence result for the bounded system to the unbounded one. For simplicity, we will explicitly note the bound index only when necessary, dropping it whenever the properties we discuss hold in both systems.

3. The confluent rewriting system

The non extensional equality rules and the rule for $\mathbf{T}$ can be turned into a confluent rewriting system by orienting them from left to right, as follows

$$\begin{array}{llcl}
(\beta) & (\lambda x:A.M)N & \longrightarrow & M[N/x] \\
(\pi_i) & \pi_i(\langle M_1, M_2\rangle) & \longrightarrow & M_i,\ \text{for } i=1,2 \\
(\rho) & Case(in^i_C(R), M_1, M_2) & \longrightarrow & M_iR,\ \text{for } i=1,2 \\
(rec) & (rec\ y:C.M)^{i+1} & \longrightarrow & M[(rec\ y:C.M)^i/y],\ \text{for } i\geq 0 \\
(Top) & M & \longrightarrow & * \text{ if } M:\mathbf{T} \text{ and } M \neq *
\end{array}$$

But when we want to turn the extensional equalities for functions and pairs into expansions, as explained very clearly by Jay (Jay92), we must be careful to avoid the following reduction loops:

$$\begin{array}{lclcl}
\lambda x.M & \leadsto & \lambda y.(\lambda x.M)y & \leadsto & \lambda y.M[y/x] =_\alpha \lambda x.M \\
\langle M, N\rangle & \leadsto & \langle \pi_1(\langle M,N\rangle), \pi_2(\langle M,N\rangle)\rangle & \leadsto & \langle M,N\rangle \\
MN & \leadsto & (\lambda x.Mx)N & \leadsto & MN \\
\pi_i(P) & \leadsto & \pi_i(\langle \pi_1(P), \pi_2(P)\rangle) & \leadsto & \pi_i(P)
\end{array}$$

To break the first two loops we must disallow expansions of terms that are already λ-abstractions or pairs:

$$\begin{array}{ll}
(\eta) & M \longrightarrow \lambda x:A.Mx \text{ if } \begin{cases} x \notin FV(M) \\ M: A\to B \text{ and } M \text{ is not a } \lambda\text{-abstraction} \end{cases} \\
(\delta) & M \longrightarrow \langle \pi_1(M), \pi_2(M)\rangle \text{ if } \{\ M: A\times B \text{ and } M \text{ is not a pair}
\end{array}$$

But this is not enough: to break the last two loops we must also forbid the η expansion of a term in a context where this term is applied to an argument, and δ expansion of a term when such a term is the argument of a projection. This means that we cannot define the one-step reduction relation $\Longrightarrow$ on terms as the least congruence on terms containing the above reductions $\longrightarrow$, as is done usually. Instead, one defines formally $M \Longrightarrow M'$ starting from $\longrightarrow$ by induction on the structure of the term. The definition is the same as a congruence closure but for the two last cases.

We will write $M \overset{\gamma_1,\ldots,\gamma_n}{\longrightarrow} M'$ if $M \overset{\gamma_i}{\longrightarrow} M'$, for some i and $\overset{\neg\gamma}{\longrightarrow}$ stands for a $\longrightarrow$ step that is not a γ step. The one-step reduction relation between terms, denoted $\Longrightarrow$ is defined as follows:

Definition 3.1. (One-step reduction)

- If $M \longrightarrow M'$, then $M \Longrightarrow M'$
- If $M \Longrightarrow M'$, then $(rec\ x:A.M)^i \Longrightarrow (rec\ x:A.M')^i$

$$\begin{array}{lll}
Case(M,N,O) \Longrightarrow Case(M',N,O) & in^1_C(M) \Longrightarrow in^1_C(M') & \langle M,N\rangle \Longrightarrow \langle M',N\rangle \\
Case(N,M,O) \Longrightarrow Case(N,M,'O) & in^2_C(M) \Longrightarrow in^2_C(M') & \langle N,M\rangle \Longrightarrow \langle N,M'\rangle \\
Case(N,O,M) \Longrightarrow Case(N,O,M') & \lambda\ x:A.M \Longrightarrow \lambda\ x:A.M' & NM \Longrightarrow NM'
\end{array}$$

- If $M \Longrightarrow M'$ but $M \overset{\neg\eta}{\longrightarrow} M'$, then $MN \Longrightarrow M'N$
- If $M \Longrightarrow M'$ but $M \overset{\neg\delta}{\longrightarrow} M'$, then $\pi_i(M) \Longrightarrow \pi_i(M')$ for $i=1,2$

Notation 3.2. The transitive and the reflexive transitive closure of $\Longrightarrow$ are noted $\Longrightarrow^+$ and $\Longrightarrow^*$ respectively. Similarly we define $\overset{\infty}{\Longrightarrow}$, $\overset{\infty}{\Longrightarrow}^+$ and $\overset{\infty}{\Longrightarrow}^*$ for the unbounded system.

We will use some standard notions from the theory of rewriting system, such as redex, normal form, confluence, weak confluence, strong normalization, etc, without explicitly redefining them here.

It is also useful to define a notion of *influential positions* of a term: informally, a position in a term is *influential* if the subterm occurring at that position cannot be expanded at the root. For example, M occurs at an influential position in the term MN, as η expansion is forbidden on M, no matters if it is a λ-abstraction or not. Obviously, a position in a term can be influential for η or for δ, but not for both. This notion can be properly formalized by induction on the structure of the terms (see (DCK93b)).

3.1. *Adequacy of expansions for extensional equalities*

First of all, it is necessary to show that the limitations imposed on the reduction system do not make us loose any valid equality. We will show that the reduction system just introduced really generates the equalities we defined for the calculus. This comes from the fact that the limitations imposed on the reductions are introduced exactly to avoid reduction loops.

Theorem 3.3. ($\Longrightarrow$ generates E) The equality E and the reflexive, symmetric and transitive closure R of $\Longrightarrow$ are the same relation.

Proof. The fact that R is included in E is evident, as all the reductions rules are derived from the equality axioms by orienting and restricting them.
What we are left to show is $E \subseteq R$. It is enough to show that whenever $M = N$ comes from a single equality axiom, we can either rewrite M to N or N to M (since R is reflexive, symmetric and transitive, the other cases will follow trivially).
The basic idea of the proof is to associate to each of these equality steps a reduction step in R. This is done in the obvious way, except in the cases that would violate one of the restrictions imposed on the expansion rules, which we will solve using exactly the reduction loop that this restriction is supposed to prevent.
Here are the problematic cases and how to deal with them. We use the usual context notation $C[M]$ to indicate a particular occurrence of a subterm M of interest in the term $C[M]$.

— $C[\lambda x.M] =_\eta C[\lambda y.(\lambda x.M)y]$. We cannot associate an η reduction to this equality, as we cannot expand something that is already an abstraction. But we can associate to it a β reduction from $C[\lambda y.(\lambda x.M)y]$ to $C[\lambda y.M[y/x]] = C[\lambda x.M]$.
— $C[\langle M, N\rangle] =_\delta \langle \pi_1(\langle M, N\rangle), \pi_2(\langle M, N\rangle)\rangle]$. We cannot expand something that is already a pair, but we can use the π_i's reduction from $\langle \pi_1(\langle M, N\rangle), \pi_2(\langle M, N\rangle)\rangle]$ to $C[\langle M, N\rangle]$.
— $C[MN] =_\eta C[(\lambda x.Mx)N]$. Here we cannot expand M, which is in an influential position, but again we can use β to go from $C[(\lambda x.Mx)N]$ to $C[MN]$ (recall that $x \notin \mathrm{FV}(M)$).
— $C[\pi_i(P)] =_\delta C[\pi_i(\langle \pi_1(P), \pi_2(P)\rangle)]$. We cannot expand P, but we can use the π_i's to go to $C[\pi_i(P)]$ from $C[\pi_i(\langle \pi_1(P), \pi_2(P)\rangle)]$.

$\square$

3.2. *Basic Properties of the Calculus*

Our calculus enjoys the Subject Reduction property, which guarantees that reduction preserves types.

Proposition 3.4. (Subject Reduction) If $\Gamma \vdash R : C$ and $R \Longrightarrow R'$, then $\Gamma \vdash R' : C$

Proof. By cases, using the following two lemmas

Lemma 3.5. If $\Gamma, x : A \vdash M : C$ and $x \notin FV(M)$, then $\Gamma \vdash M : C$.

Lemma 3.6. If $\Gamma, x : A \vdash M : C$ and $\Gamma \vdash N : A$, then $\Gamma \vdash M[N/x] : C$.

$\square$

Another remarkable property of this calculus can be stated as follows:

Lemma 3.7. If M is in normal form w.r.t. the system without the η, δ and Top rules and $M \overset{\eta,\delta,Top}{\Longrightarrow} R$, then R is in normal form w.r.t. the system without η, δ and Top.

Proof. Suppose M has no β, π_i, ρ or rec redexes. Notice first that a ρ redex cannot be created in R as there is no way to introduce an in^i expression using the η, δ and Top rules. The same for rec. We are left with the following two cases:

— Suppose R has a β redex. Then $R \equiv C[(\lambda x.P)N]$ and since M contains no β redexes, we have necessarily $M \equiv C[SN]$, $P \equiv Sx$ and $S \xrightarrow{\eta} \lambda x.Sx$. But this is not possible because η expansions are not allowed on terms appearing in influential positions for η.
— Suppose R has a π_i redex. Then $R \equiv C[\pi_i(\langle M, N\rangle)]$ and since M contains no π_i's redexes, we have necessarily $M \equiv C[\pi_i(T)]$, $M \equiv \pi_1(T)$, $N \equiv \pi_2(T)$ and $T \xrightarrow{\delta} \langle \pi_1(T), \pi_2(T)\rangle$. But this is not possible because δ expansions are not allowed on terms appearing in influential positions for δ.

□

Corollary 3.8. If M is in normal form with respect to the system without the η, δ and Top rules, then the (η, δ, Top)-normal form of M is in normal form.

4. Weak Confluence

In this section we set off to prove that the reduction system proposed above is actually weakly confluent, i.e. that whenever $M' \Longleftarrow M \Longrightarrow M''$ we can find a term M''' s.t. $M' \Longrightarrow^* M''' \;{}^*\!\Longleftarrow M''$. The proof is fairly more complex here than in the case of λ-calculus where extensional equalities are interpreted as contractions, and this is due to the fact that the reduction relation $\Longrightarrow$ introduced above *is not a congruence on terms*.

4.1. *Some difficulties*

In particular, in the simply typed $\lambda-$calculus whenever $M \Longrightarrow^* M'$ then $\pi_i(M) \Longrightarrow^* \pi_i(M')$, and if also $N \Longrightarrow^* N'$, then $MN \Longrightarrow^* M'N'$, but this is no longer true now: we have $x : A \to B \Longrightarrow \lambda z : A.xz$, but xN cannot reduce to $(\lambda z : A.xz)N$.

These properties still hold for those reduction sequences $M \Longrightarrow^* M'$ that do not involve expansions at the root:

Remark 4.1.

— Let $M \equiv M_0 \Longrightarrow M_1 \Longrightarrow \ldots \Longrightarrow M_{n-1} \Longrightarrow M_n \equiv M'$ be a reduction sequence and let $N \Longrightarrow^* N'$, where in the first reduction sequence there are no expansions applied at the root positions of the M_i's. Then, $MN \Longrightarrow^* M'N'$.
— Let $M \equiv M_0 \Longrightarrow M_1 \Longrightarrow \ldots \Longrightarrow M_{n-1} \Longrightarrow M_n \equiv M'$ be a reduction sequence where none of the $M_i's$ is expanded at the root. Then $\pi_i(M) \Longrightarrow^* \pi_i(M')$, for $i = 1, 2$.

4.2. *Solving Critical Pairs*

In this calculus, it is no longer true that reduction is stable by substitution, as in the traditional λ-calculus:

Remark 4.2.
If $P \Longrightarrow P'$ and $N \Longrightarrow N'$, then $P[N/x] \Longrightarrow^* P'[N/x]$ and $P[N/x] \Longrightarrow^* P[N'/x]$ are not necessarily true.

Indeed, $x : A \to B \Longrightarrow \lambda z : A.xz$, but $x[\lambda y : A.w/x] = \lambda y : A.w$ cannot reduce in our system to $\lambda z : A.(\lambda y : A.w)z = \lambda z : A.xz[\lambda y : A.w/x]$, and $(yM)[x/y] = xM$ cannot reduce to $(\lambda z : A.xz)M = (yM)[\lambda z : A.xz/y]$.

We can prove some weaker properties: if $P \Longrightarrow P'$, then $P[N/x]$ and $P'[N/x]$ have a common reduct (Lemma 4.5), and similarly $P[N/x]$ and $P[N'/x]$ when $N \Longrightarrow N'$ (Lemma 4.6). This suffices for our purpose of proving weak confluence of the reduction system.

First of all it is useful to recall here a basic property of substitutions that do hold in our calculus.

Lemma 4.3. If $x \not\equiv y$ and $x \notin FV(L)$, then

$$M[N/x][L/y] = M[L/y][N[L/y]/x]$$

Lemma 4.4. If $P \stackrel{\eta,\delta,Top}{\longrightarrow} P'$, then $P[N/x] \Longrightarrow^* P'[N/x]$ or $P'[N/x] \Longrightarrow^* P[N/x]$. Moreover, if the expansion does not take place at the root of P, then there are no expansions at root positions in the sequences $P[N/x] \Longrightarrow^* P'[N/x]$ and $P'[N/x] \Longrightarrow^* P[N/x]$.

Proof.

— $P \stackrel{\eta}{\longrightarrow} \lambda z.Pz$. Then P is not a $\lambda-$abstraction, $P'[N/x] = \lambda z.P[N/x]z$ and there are two possible cases:

- If $P[N/x]$ is not a $\lambda-$abstraction, $P[N/x] \stackrel{\eta}{\longrightarrow} \lambda z.P[N/x]z$ since P is of type $\rightarrow$ and so $P[N/x]$ is also of type $\rightarrow$ by lemma 3.6.
- If $P[N/x]$ is a $\lambda-$abstraction, then $P \equiv x$, $N \equiv \lambda y.N'$ and:
  $(\lambda z.xz)[\lambda y.N'/x] = \lambda z.(\lambda y.N')z \stackrel{\beta}{\longrightarrow} \lambda z.(N'[z/y]) =_\alpha \lambda y.N' = x[\lambda y.N'/x]$.

— $P \stackrel{\delta}{\longrightarrow} \langle \pi_1(P), \pi_2(P)\rangle$. Then P is not a pair, $P'[N/x] = \langle \pi_1(P[N/x]), \pi_2(P[N/x])\rangle$ and there are two possible cases:

- If $P[N/x]$ is not a pair, $P[N/x] \stackrel{\delta}{\longrightarrow} \langle \pi_1(P[N/x]), \pi_2(P[N/x])\rangle$ since P is of type $\times$ and so $P[N/x]$ is also of type $\times$ by lemma 3.6.
- If $P[N/x]$ is a pair, then $P \equiv x$ and $N \equiv \langle N_1, N_2\rangle$ and:

$$\begin{array}{ll} \langle \pi_1(x), \pi_2(x)\rangle[\langle N_1, N_2\rangle/x] & x[\langle N_1, N_2\rangle/x] \\ = & = \\ \langle \pi_1(\langle N_1, N_2\rangle), \pi_2(\langle N_1, N_2\rangle)\rangle \stackrel{\pi_1}{\Longrightarrow} \langle N_1, \pi_2(\langle N_1, N_2\rangle)\rangle \stackrel{\pi_2}{\Longrightarrow} & \langle N_1, N_2\rangle \end{array}$$

— $P \stackrel{Top}{\longrightarrow} *$. Then $P[N/x] \stackrel{Top}{\longrightarrow} * = *[N/x]$ since P is of type $\mathbf{T}$ and so $P[N/x]$ is also of type $\mathbf{T}$ by lemma 3.6.

$\square$

Using the previous Lemma, we can precisely describe the interaction between reductions and substitutions.

Lemma 4.5. (Substitution Lemma (i))
If $P \Longrightarrow P'$, then $P[N/x] \Longrightarrow^* P'[N/x]$ or $P'[N/x] \Longrightarrow^* P[N/x]$. Moreover, if no expansion take place at the root position of P, then there are no expansions at root positions in the reduction sequences $P[N/x] \Longrightarrow^* P'[N/x]$ and $P'[N/x] \Longrightarrow^* P[N/x]$.

Proof. The property is shown by induction on the structure of P. We only show the argument of the proof for $Case$ as an illustration.
$P \equiv Case(Q, M_1, M_2)$. If $P \stackrel{\eta,\delta,Top}{\longrightarrow} P'$ the property holds by lemma 4.4. If not, there are five possibilities:

— $Q \equiv in^i_{B_1+B_2}(R)$ and $Case(in^i_{B_1+B_2}(R), M_1, M_2) \stackrel{\rho}{\Longrightarrow}^* M_iR$. Since

$$Case(in^i_{B_1+B_2}(R), M_1, M_2)[N/x] = Case(in^i_{B_1+B_2}(R[N/x]), M_1[N/x], M_2[N/x])$$

and this last term reduces by a ρ-rule to $M_i[N/x]R[N/x] = (M_iR)[N/x]$ the property holds.

— $P' \equiv Case(Q', M_1, M_2)$, where $Q \Longrightarrow Q'$.
By the induction hypothesis $Q[N/x] \Longrightarrow^* Q'[N/x]$ or $Q'[N/x] \Longrightarrow^* Q[N/x]$.
In the first case

$$\begin{array}{ccc} Case(Q, M_1, M_2)[N/x] & & Case(Q', M_1, M_2)[N/x] \\ = & & = \\ Case(Q[N/x], M_1[N/x], M_2[N/x]) & \Longrightarrow^* & Case(Q'[N/x], M_1[N/x], M_2[N/x]) \end{array}$$

In the second case $Case(Q', M_1, M_2)[N/x] \Longrightarrow^* Case(Q, M_1, M_2)[N/x]$.

— $P' \equiv Case(Q, M_1', M_2)$, where $M_1 \Longrightarrow M_1'$. By the induction hypothesis we have the reduction sequences $M_1[N/x] \Longrightarrow^* M_1'[N/x]$ or $M_1'[N/x] \Longrightarrow^* M_1[N/x]$.
In the first case

$$\begin{array}{ccc} Case(Q, M_1, M_2)[N/x] & & Case(Q, M_1', M_2)[N/x] \\ = & & = \\ Case(Q[N/x], M_1[N/x], M_2[N/x]) & \Longrightarrow^* & Case(Q[N/x], M_1'[N/x], M_2[N/x]) \end{array}$$

In the second case $Case(Q, M_1', M_2)[N/x] \Longrightarrow^* Case(Q, M_1, M_2)[N/x]$.

— $P' \equiv Case(Q, M_1, M_2')$, where $M_2 \Longrightarrow M_2'$. By the induction hypothesis we have the reduction sequences $M_2[N/x] \Longrightarrow^* M_2'[N/x]$ or $M_2'[N/x] \Longrightarrow^* M_2[N/x]$.
In the first case

$$\begin{array}{ccc} Case(Q, M_1, M_2)[N/x] & & Case(Q, M_1, M_2')[N/x] \\ = & & = \\ Case(Q[N/x], M_1[N/x], M_2[N/x]) & \Longrightarrow^* & Case(Q[N/x], M_1[N/x], M_2'[N/x]) \end{array}$$

In the second case $Case(Q, M_1, M_2')[N/x] \Longrightarrow^* Case(Q, M_1, M_2)[N/x]$.

□

Lemma 4.6. (Substitution Lemma (ii))
If $N \stackrel{R}{\Longrightarrow} N'$, then $M[N/x] \Longrightarrow^* M'' \; {}^*\!\!\Longleftarrow M[N'/x]$ for some term M''. These reduction sequences contain expansions at the root only if $M \equiv x$ and R is an expansion applied at the root of N.

Proof. We will show that $M[N/x] \Longrightarrow^* M'' \; {}^*\!\!\Longleftarrow M[N'/x]$ for some term M'' *and* that these reduction sequences contain expansions at the root only if $M \equiv x$ and R is an expansion applied at the root of N.
This is a very common lemma in the theory of λ-calculus, where the term M'' is always $M[N'/x]$ and the proof is straightforward context closure of the reduction R. Here the conditions imposed on the expansion rules make it necessary to state the lemma this way. Effectively, the only interesting cases of the proof are the ones for application and projections, where we cannot always apply context closure for the reduction R, and have to make some steps backwards from $M[N'/x]$ to $M[N/x]$.

Notice that every time the required reductions are built by context closure, there is no rule applied at the root and we state this fact here once for all. We proceed by induction on M:

— $M \equiv x$
$M[N/x] = N \stackrel{R}{\Longrightarrow} N' = M[N'/x]$ (in this case our M'' is N')
— $M \equiv y \not\equiv x$ or $M \equiv * : \mathbf{T}$
Then $M[N/x] = M = M[N'/x]$ (in this case our M'' is M)

— $M \equiv (M_1 M_2)$
By the induction hypothesis we have M_1'' and M_2'' such that

$$M_1[N/x] \Longrightarrow^* M_1'' \,{}^*\!\Longleftarrow M_1[N'/x], \text{ and } M_2[N/x] \Longrightarrow^* M_2'' \,{}^*\!\Longleftarrow M_2[N'/x].$$

Here M_1 *is in an influential position for* η, so we have to be careful about the reductions occurring in $M_1[N/x] \Longrightarrow^* M_1'' \,{}^*\!\Longleftarrow M_1[N'/x]$. We have the following cases:

– If $M_1 \not\equiv x$, or R is not an expansion at the root of N, we know by inductive hypothesis that the reductions $M_1[N/x] \Longrightarrow^* M_1'' \,{}^*\!\Longleftarrow M_1[N'/x]$ do not contain any expansions, and in particular no η rule, at the root position, so we can apply context closure for application and get

$$\begin{array}{ccccc} (M_1M_2)[N/x] & & & & (M_1M_2)[N'/x] \\ = & & & & = \\ (M_1[N/x]M_2[N/x]) & \Longrightarrow^* & (M_1''M_2'') & {}^*\!\Longleftarrow & (M_1[N'/x]M_2[N'/x]). \end{array}$$

– If $M_1 \equiv x$, and the expansion rule R is η at the root of N, then $N' \equiv \lambda z.Nz$ and we can close our diagram as follows

$$\begin{array}{ccc} (x[N/x]M_2[N/x]) & & (x[\lambda z.Nz/x]M_2[\lambda z.Nz/x]) \\ = & & = \\ (NM_2[N/x]) & & ((\lambda z.Nz)M_2[\lambda z.Nz/x]) \\ \Downarrow & & \Downarrow \\ (NM_2'') & \xLongleftarrow{\beta} & (\lambda z.Nz)M_2'' \end{array}$$

Here, the vertical reductions are built by context closure, while the horizontal one is a β, so no expansion rule is applied at the root in the overall reduction sequence.

— $M \equiv \lambda z : A.M_1$
If $x \neq z$, the result follows from $(\lambda z : A.M_1)[N/x] = \lambda z : A.M_1 = (\lambda z : A.M_1)[N'/x]$. Otherwise, there is a term M_1'' such that $M_1[N/x] \Longrightarrow^* M_1'' \,{}^*\!\Longleftarrow M_1[N'/x]$ by the induction hypothesis, so we can apply the context closure rule for abstraction and get that

$$\begin{array}{ccccc} (\lambda z : A.M_1)[N/x] & & & & (\lambda z.M_1)[N'/x] \\ = & & & & = \\ (\lambda z : A.M_1[N/x]) & \Longrightarrow^* & \lambda z : A.M_1'' & {}^*\!\Longleftarrow & (\lambda z : A.M_1[N'/x]) \end{array}$$

— $M \equiv \pi_i(M_1)$
By the induction hypothesis we can find a term M_1'' such that $M_1[N/x] \Longrightarrow^* M_1''$ and $M_1[N'/x] \Longrightarrow^* M_1''$. Here M_1 *is in an influential position for* δ, so we have to be careful about the reductions occurring in $M_1[N/x] \Longrightarrow^* M_1'' \,{}^*\!\Longleftarrow M_1[N'/x]$. We have the following cases:

– If $M_1 \not\equiv x$, or R is not an expansion at the root of N, we know by inductive hypothesis that the reductions $M_1[N/x] \Longrightarrow^* M_1'' \,{}^*\!\Longleftarrow M_1[N'/x]$ do not contain any expansions, and in particular no δ rule, at the root position, so we can apply context closure for projections and get

$$\pi_i(M_1)[N/x] \Longrightarrow^* \pi_i(M_1'') \,{}^*\!\Longleftarrow \pi_i(M_1)[N'/x].$$

– If $M_1 \equiv x$, and the expansion rule R is δ at the root of N, then $N' \equiv \langle \pi_1(N), \pi_2(N) \rangle$ and we can close our diagram as follows

$$\begin{array}{ccc} \pi_i(x[N/x]) & & \pi_i(x[\langle \pi_1 N, \pi_2 N \rangle/x]) \\ = & & = \\ \pi_i(N) & \xLongleftarrow{\pi_i} & \pi_i\langle \pi_1 N, \pi_2 N \rangle \end{array}$$

Here, the vertical reductions are built by context closure, while the horizontal one is a π, so no expansion rule is applied at the root in the overall reduction sequence.

— $M \equiv \langle M_1, M_2\rangle$
By the induction hypothesis we have $M_1[N/x] \Longrightarrow^* M_1'' \ {}^*\!\!\Longleftarrow M_1[N'/x]$ for some term M_1'', and we have $M_2[N/x] \Longrightarrow^* M_2'' \ {}^*\!\!\Longleftarrow M_2[N'/x]$ for some term M_2''. So, we can apply the context closure rule for application and get that

$$\begin{array}{ccc} (\langle M_1, M_2\rangle)[N/x] & & (\langle M_1, M_2\rangle)[N'/x] \\ = & & = \\ \langle M_1[N/x], M_2[N/x]\rangle & & \langle M_1[N'/x], M_2[N'/x]\rangle \\ {}^*\Downarrow & & \Downarrow^* \\ \langle M_1'', M_2''\rangle & = & \langle M_1'', M_2''\rangle \end{array}$$

— $M \equiv in^i_C(M_1)$
We find by the induction hypothesis a term M_1'' such that $M_1[N/x] \Longrightarrow^* M_1''$ and $M_1[N'/x] \Longrightarrow^* M_1''$, so we can apply the context closure rule for in^i and get that

$$\begin{array}{ccccc} in^i_C(M_1[N/x]) & & & & in^i_C(M_1[N'/x]) \\ = & & & & = \\ in^i_C(M_1)[N/x] & \Longrightarrow^* & in^i_C(M_1'') & {}^*\!\!\Longleftarrow & in^i_C(M_1)[N'/x] \end{array}$$

— $M \equiv Case(P, M_1, M_2)$
By the induction hypothesis we have $P[N/x] \Longrightarrow^* P'' \ {}^*\!\!\Longleftarrow P[N'/x]$ for some term P'', and we have $M_1[N/x] \Longrightarrow^* M_1'' \ {}^*\!\!\Longleftarrow M_1[N'/x]$ and $M_2[N/x] \Longrightarrow^* M_2'' \ {}^*\!\!\Longleftarrow M_2[N'/x]$ for some terms M_1'' and M_2''. So, we can apply the context closure rule for $Case$ and get that

$$\begin{array}{ccc} Case(P,M_1,M_2)[N/x] & & Case(P,M_1,M_2)[N'/x] \\ = & & = \\ Case(P[N/x],(M_1)[N/x],(M_2)[N/x]) & & Case(P[N'/x],(M_1)[N'/x],(M_2)[N'/x]) \\ {}^*\Downarrow & & \Downarrow^* \\ Case(P'', M_1'', M_2'') & = & Case(P'', M_1'', M_2'') \end{array}$$

— $M \equiv (rec\, z : A.M_1)^i$
We assume $z \not\equiv x$ (otherwise the result trivially holds). We find by the induction hypothesis a term M_1'' such that $M_1[N/x] \Longrightarrow^* M_1'' \ {}^*\!\!\Longleftarrow M_1[N'/x]$. so we can apply the context closure rule for rec and get that

$$\begin{array}{ccccc} (rec\, z : A.M_1)^i[N/x] & & & & (rec\, z : A.M_1)^i[N'/x] \\ = & & & & = \\ (rec\, z : A.M_1[N/x])^i & \Longrightarrow^* & (rec\, z : A.M_1'')^i & {}^*\!\!\Longleftarrow & (rec\, z : A.M_1[N'/x])^i \end{array}$$

□

Example 4.7. Take $M = \langle xy, x\rangle$, $N = w$ and $N' = \lambda z : A.wz$. Then

$$M[N/x] = \langle wy, w\rangle \Longrightarrow \langle wy, \lambda z : A.wz\rangle \Longleftarrow \langle(\lambda z : A.wz)y, \lambda z : A.wz\rangle = M[N'/x]$$

Looking carefully through the proof of the previous Lemma 4.6, one can see that the only cases where it is needed to apply a reverse reduction are those corresponding to an expansion rule applied at the root of N and to the presence in M of some free occurrences of x in *influential* positions. So, we can also state the following

Corollary 4.8. (Reverse reductions) Let $N \stackrel{R}{\Longrightarrow} N'$. In case R is not an expansion rule applied at the root of N (an *external expansion rule*) or x does not occur at an influential position in M, then $M[N/x] \Longrightarrow^* M[N'/x]$

Lemma 4.5 and 4.6 suffice to prove that all critical pairs arising from a term M by a β-reduction and another reduction rule can be solved. We can then state the following:

Proposition 4.9. (Critical Pairs are solvable)
If $M \rightarrow M'$ and $M \Longrightarrow M''$, then $\exists R$ such that $M' \Longrightarrow^* R$ and $M'' \Longrightarrow^* R$.

Proof. One considers all cases of reduction from M to M'. We show here only some *interesting* cases, since confluence for the other cases is shown in many of the mentioned references, and full details are given in (DCK93b).

1. $M \xrightarrow{\beta} M'$. Thus $M \equiv (\lambda x.P)N$.

1.1. If $M \Longrightarrow M''$ is internal, there are two cases:

— $P \Longrightarrow P'$

$$\begin{array}{ccc} (\lambda x.P)N & \Longrightarrow & (\lambda x.P')N \\ \beta\big\downarrow & & \beta\big\downarrow \\ P[N/x] & & P'[N/x] \end{array}$$

By lemma 4.5 we have $P[N/x] \Longrightarrow^* P'[N/x]$ or $P'[N/x] \Longrightarrow^* P[N/x]$.

— $N \Longrightarrow N'$

$$\begin{array}{ccc} (\lambda x.P)N & \Longrightarrow & (\lambda x.P)N' \\ \beta\big\downarrow & & \beta\big\downarrow \\ P[N/x] & & P[N'/x] \end{array}$$

By lemma 4.6 there is a term R such that $P[N/x] \Longrightarrow^* R$ and $P[N'/x] \Longrightarrow^* R$.

1.2. If $M \Longrightarrow M''$ is external the interesting cases involve η and δ :

1.2.1. $M \equiv (\lambda x.P)N \xrightarrow{\eta} \lambda y.((\lambda x.P)N)y \equiv M''$

— If $P[N/x]$ is not a $\lambda-$abstraction:

$$\begin{array}{ccc} (\lambda x.P)N & \xrightarrow{\eta} & \lambda y.((\lambda x.P)N)y \\ \beta\big\downarrow & & \beta\Big\Downarrow \\ P[N/x] & \xrightarrow{\eta} & \lambda y.P[N/x]y \end{array}$$

— If $P[N/x]$ is a $\lambda-$abstraction we have two cases:

– If P is a $\lambda-$abstraction:

$$\begin{tikzcd}
(\lambda x.(\lambda z.P'))N \arrow[r, "\eta"] \arrow[ddd, "\beta"'] & \lambda y.((\lambda x.(\lambda z.P'))N)y \arrow[d, Rightarrow, "\beta"'] \\
 & \lambda y.((\lambda z.P')[N/x])y \arrow[d, phantom, "="] \\
 & \lambda y.(\lambda z.(P'[N/x])y) \arrow[dd, Rightarrow, "\beta"'] \\
(\lambda z.P')[N/x] \arrow[d, phantom, "="] & \\
\lambda z.(P'[N/x]) \arrow[r, phantom, "="] & \lambda y.P'[N/x][y/z]
\end{tikzcd}$$

– If $P = x$ and N is a $\lambda-$abstraction $\lambda z.N'$:

$$\begin{tikzcd}
(\lambda x.x)\lambda z.N' \arrow[r, "\eta"] \arrow[dd, "\beta"'] & (\lambda y.((\lambda x.x)\lambda z.N')y) \arrow[d, Rightarrow, "\beta"'] \\
 & \lambda y.(\lambda z.N')y \arrow[d, Rightarrow, "\beta"'] \\
\lambda z.N' \arrow[r, phantom, "="] & \lambda y.N'[y/z]
\end{tikzcd}$$

1.2.2. $M \equiv (\lambda x.P)N \xrightarrow{\delta} \langle \pi_1((\lambda x.P)N), \pi_2((\lambda x.P)N)\rangle \equiv M''$

— If $P[N/x]$ is not a pair we have:

$$\begin{tikzcd}
(\lambda x.P)N \arrow[r, "\delta"] \arrow[dd, "\beta"'] & \langle \pi_1((\lambda x.P)N), \pi_2((\lambda x.P)N)\rangle \arrow[d, Rightarrow, "\beta"'] \\
 & \langle \pi_1(P[N/x]), \pi_2((\lambda x.P)N)\rangle \arrow[d, Rightarrow, "\beta"'] \\
P[N/x] \arrow[r, "\delta"] & \langle \pi_1(P[N/x]), \pi_2(P[N/x])\rangle
\end{tikzcd}$$

— If $P[N/x]$ is a pair we have two more cases:

– P is also a pair $\langle P_1, P_2\rangle$:

$$
\begin{tikzcd}
(\lambda x.\langle P_1, P_2\rangle)N \arrow[r, "\delta"] \arrow[dddd, "\beta"'] & \langle \pi_1((\lambda x.\langle P_1, P_2\rangle)N), \pi_2((\lambda x.\langle P_1, P_2\rangle)N)\rangle \arrow[d, Rightarrow, "\beta"'] \\
 & \langle \pi_1(\langle P_1, P_2\rangle[N/x]), \pi_2((\lambda x.\langle P_1, P_2\rangle)N)\rangle \arrow[d, Rightarrow, "\beta"'] \\
 & \langle \pi_1(\langle P_1, P_2\rangle[N/x]), \pi_2(\langle P_1, P_2\rangle[N/x])\rangle \arrow[d, Rightarrow, "\pi_1"'] \\
 & \langle P_1[N/x], \pi_2(\langle P_1, P_2\rangle[N/x])\rangle \arrow[dl, Rightarrow, "\pi_2"'] \\
\langle P_1[N/x], P_2[N/x]\rangle &
\end{tikzcd}
$$

– $P = x$ and N is a pair $\langle N_1, N_2\rangle$:

$$
\begin{tikzcd}
(\lambda x.x)\langle N_1, N_2\rangle \arrow[r, "\delta"] \arrow[dddd, "\beta"'] & \langle \pi_1((\lambda x.x)\langle N_1, N_2\rangle), \pi_2((\lambda x.x)\langle N_1, N_2\rangle)\rangle \arrow[d, Rightarrow, "\beta"'] \\
 & \langle \pi_1(\langle N_1, N_2\rangle), \pi_2((\lambda x.x)\langle N_1, N_2\rangle)\rangle \arrow[d, Rightarrow, "\beta"'] \\
 & \langle \pi_1(\langle N_1, N_2\rangle), \pi_2(\langle N_1, N_2\rangle)\rangle \arrow[d, Rightarrow, "\pi_1"'] \\
 & \langle N_1, \pi_2(\langle N_1, N_2\rangle)\rangle \arrow[dl, Rightarrow, "\pi_2"'] \\
\langle N_1, N_2\rangle &
\end{tikzcd}
$$

2. $M \xrightarrow{\eta} M'$.

2.1. If $M \Longrightarrow M''$ is internal, then the same reduction can be performed on $\lambda z.Mz$, and the outermost term constructor of M and M'' does not change, so an expansion is still possible on M'', and we can generally close the diagram as follows:

$$
\begin{tikzcd}
M \arrow[r, Rightarrow] \arrow[d, "\eta"'] & M'' \arrow[d, "\eta"'] \\
\lambda z.Mz \arrow[r, Rightarrow] & \lambda z.(M'')z
\end{tikzcd}
$$

2.2. If $M \Longrightarrow M''$ is external, the interesting cases are:

2.2.1. $M \xrightarrow{\pi_i} M''$. Then $M \equiv \pi_i(\langle M_1, M_2\rangle)$ and there are two cases:

— If M_i is not a $\lambda-$abstraction, the diagram looks like:

$$
\begin{tikzcd}
\pi_i(\langle M_1, M_2\rangle) \arrow[r, "\pi_i"] \arrow[d, "\eta"] & M_i \arrow[d, "\eta"] \\
\lambda z.\pi_i(\langle M_1, M_2\rangle)z \arrow[r, Rightarrow, "\pi_i"] & \lambda z.M_i z
\end{tikzcd}
$$

— If M_i is a $\lambda-$abstraction $\lambda y.M_i^{'}$, the diagram looks like:

$$\begin{array}{ccc}
\pi_i(\langle M_1, M_2\rangle) & \xrightarrow{\pi_i} & \lambda y.M_i^{'} \\
\downarrow \eta & & \\
\lambda z.\pi_i(\langle M_3, M_2\rangle)z & & \\
\pi_i \Downarrow & & \\
\lambda z.(\lambda y.M_i^{'})z & & \\
\beta \Downarrow & & \\
\lambda z.(M_i^{'}[z/y]) & & =
\end{array}$$

2.2.2. $M \xrightarrow{rec} M''$. Then $M = (rec\, y.M_1)^i$ and there are two possible cases:

— If M_1 is not a $\lambda-$abstraction:

$$\begin{array}{ccc}
(rec\, y.M_1)^i & \xrightarrow{rec} & M_1[(rec\, y.M_1)^{i-1}/y] \\
\eta\downarrow & & \eta\downarrow \\
\lambda z.(rec\, y.M_1)^i z & \overset{rec}{\Longrightarrow} & \lambda z.(M_1[(rec\, y.M_1)^{i-1}/y])z
\end{array}$$

— If $M_1 \equiv \lambda w.M_1^{'}$:

$$\begin{array}{ccc}
(rec\, y.(\lambda w.M_1^{'}))^i & \xrightarrow{rec} & (\lambda w.M_1^{'})[(rec\, y.(\lambda w.M_1^{'}))^{i-1}/y] \\
\eta\downarrow & & \\
\lambda z.(rec\, y.\lambda w.M_1^{'})^i z & & \\
rec \Downarrow & & \\
\lambda z.(\lambda w.M_1^{'}[(rec\, y.(\lambda w.M_1^{'}))^{i-1}/y])z & & \\
\beta \Downarrow & & \\
\lambda z.(M_1^{'}[(rec\, y.(\lambda w.M_1^{'}))^{i-1}/y][z/w] & & =
\end{array}$$

3. $M \xrightarrow{\delta} M'$.

3.1. If $M \Longrightarrow M''$ is internal, then the same reduction can be performed on $\langle \pi_1(M), \pi_2(M)\rangle$, and the outermost term constructor of M and M'' does not change, so an expansion is still possible on M'', and we can generally close the diagram as follows:

$$\begin{array}{ccc}
M & \Longrightarrow & M^{''} \\
\delta\downarrow & & \delta\downarrow \\
\langle \pi_1(M), \pi_2(M)\rangle & \Longrightarrow & \langle \pi_1(M), \pi_2(M^{''})\rangle
\end{array}$$

3.2. If $M \Longrightarrow M''$ is external the interesting cases are:

3.2.1. $M \xrightarrow{\pi_i} M''$. Then $M \equiv \pi_i(\langle M_1, M_2\rangle)$.

— If M_i is not a pair:

$$
\begin{tikzcd}
\pi_i(\langle M_1, M_2\rangle) \arrow[r, "\delta"] \arrow[dd, "\pi_i"'] & \langle \pi_1(\pi_i(\langle M_1, M_2\rangle)), \pi_2(\pi_i(\langle M_1, M_2\rangle))\rangle \arrow[d, Rightarrow, "\pi_i"'] \\
 & \langle \pi_1(M_i), \pi_2(\pi_i(\langle M_1, M_2\rangle))\rangle \arrow[d, Rightarrow, "\pi_i"'] \\
M_i \arrow[r, "\delta"] & \langle \pi_1(M_i), \pi_2(M_i)\rangle
\end{tikzcd}
$$

— If M_i is a pair $\langle P_1, P_2\rangle$:

$$
\begin{tikzcd}
\pi_1(\langle M_1, M_2\rangle) \arrow[r, "\delta"] \arrow[dddd, "\pi_i"'] & \langle \pi_1(\pi_i(\langle M_1, M_2\rangle)), \pi_2(\pi_i(\langle M_1, M_2\rangle))\rangle \arrow[d, Rightarrow, "\pi_i"'] \\
 & \langle \pi_1(\langle P_1, P_2\rangle), \pi_2(\pi_1(\langle M_1, M_2\rangle))\rangle \arrow[d, Rightarrow, "\pi_i"'] \\
 & \langle \pi_1(\langle P_1, P_2\rangle), \pi_2(\langle P_1, P_2\rangle)\rangle \arrow[d, Rightarrow, "\pi_1"'] \\
 & \\
\langle P_1, P_2\rangle & \langle P_1, \pi_2(\langle P_1, P_2\rangle)\rangle \arrow[l, Rightarrow, "\pi_2"']
\end{tikzcd}
$$

3.2.2. $M \equiv (rec\ y : C.P)^i \to P[(rec\ y : C.P)^{i-1}/y] \equiv M''$ where P is a pair $\langle P_1, P_2\rangle$.

$$
\begin{tikzcd}
M \equiv (rec\ y : C.\langle P_1, P_2\rangle)^i \arrow[r, Rightarrow] \arrow[d, "\delta"'] & \langle P_1, P_2\rangle[M/y] \\
\langle \pi_1(M), \pi_2(M)\rangle \arrow[d, Rightarrow, "rec"] & \\
\langle \pi_1(\langle P_1, P_2\rangle[M/y]), \pi_2(M)\rangle \arrow[d, Rightarrow, "rec"] & \\
\langle \pi_1(\langle P_1, P_2\rangle[M/y]), \pi_2(\langle P_1, P_2\rangle[M/y])\rangle \arrow[d, Rightarrow, "\pi_1"] & \\
\langle P_1[M/y], \pi_2(\langle P_1, P_2\rangle[M/y])\rangle \arrow[d, Rightarrow, "\pi_2"] & \\
\langle P_1[M/y], P_2[M/y]\rangle & =
\end{tikzcd}
$$

$\square$

4.3. *From Solved Critical Pairs to Full Weak Confluence*

It is to be noted that the solvability of critical pairs we just proved as Proposition 4.9 does not allow us to deduce the weak confluence of the calculus via the famous Knuth-Bendix Critical Pairs Lemma. That Lemma only deals with algebraic rewrite systems, and cannot be used for our calculus, that has the higher order rewrite rule β. We need to prove local confluence explicitly, and to do so the following remark is useful.

Remark 4.10. (Expansion rules) In case the two reductions $M' \longleftarrow M \Longrightarrow M''$ do not involve η (resp. δ) rules applied at the root positions of M, it is possible to close the diagram without using η (resp. δ) rules at the root, except in the three cases shown below: external π_i's and internal η, external β and internal δ. Notice that M is not a λ- abstraction in the first diagram, N is not a λ- abstraction in the second and $M[N/x]$ is not a pair in the third one.

$$\begin{tikzcd}
\pi_1(\langle M,N\rangle) \arrow[r, Rightarrow, "\eta"] \arrow[d, "\pi"'] & \pi_1(\langle \lambda x.Mx, N\rangle) \arrow[d, Rightarrow, "\pi"] \\
M \arrow[r, Rightarrow, "\eta"] & \lambda x.Mx
\end{tikzcd}
\qquad
\begin{tikzcd}
\pi_2(\langle M,N\rangle) \arrow[r, Rightarrow, "\eta"] \arrow[d, "\pi"'] & \pi_2(\langle M, \lambda x.Nx\rangle) \arrow[d, Rightarrow, "\pi"] \\
N \arrow[r, Rightarrow, "\eta"] & \lambda x.Nx
\end{tikzcd}$$

$$\begin{tikzcd}
(\lambda x:A.M)N \arrow[r, Rightarrow, "\delta"] \arrow[d, "\beta"'] & (\lambda x:A.\langle \pi_1(M), \pi_2(M)\rangle)N \arrow[d, Rightarrow, "\beta"] \\
M[N/x] \arrow[r, Rightarrow, "\delta"] & \langle \pi_1(M[N/x]), \pi_2(M[N/x])\rangle
\end{tikzcd}$$

With this additional knowledge, we can prove that $\Longrightarrow$ is actually weakly confluent.

Theorem 4.11. (Weak Confluence) If $M' \Longleftarrow M \Longrightarrow M''$ then there exist a term M''' such that $M' \Longrightarrow^* M''' \ {}^*\!\!\Longleftarrow M''$ (i.e. the reduction relation $\Longrightarrow$ is weakly confluent). Furthermore, if the reductions in $M' \Longleftarrow M \Longrightarrow M''$ do not contain η (resp. δ) rules applied at the root of M, it is possible also to close the diagram without applying η (resp. δ) rules at the root, except in the cases shown in the previous Remark 4.10.

Proof. We will prove that there exists a term M''' such that $M' \Longrightarrow^* M''' \ {}^*\!\!\Longleftarrow M''$, by induction on the derivation of $M \Longrightarrow M'$. First of all, we remark that if one of the two one-step reductions $M \Longrightarrow M'$ and $M \Longrightarrow M''$ is actually an external reduction $M \xrightarrow{M} {}'$ and $M \xrightarrow{M} {}''$, then the result comes directly from Proposition 4.9. So we will need to consider in the following only the cases where both reductions are internal reductions.

We proceed now by cases on the last rule used to derive $M \Longrightarrow M'$.

— $M \equiv (M_1 M_2) \Longrightarrow (M_1' M_2) \equiv M'$ comes from $M_1 \Longrightarrow M_1'$. In this case, the η rule cannot be applied at the root position of M_1 because M_1 is evaluated. Then we have two cases:

- The reduction $M \equiv (M_1 M_2) \Longrightarrow (M_1'' M_2) \equiv M''$ comes from a one-step reduction $M_1 \Longrightarrow M_1''$. Now we have to consider two cases:
  - $M_1' \Longleftarrow M_1 \Longrightarrow M_1''$ is not one of the exceptional cases for η of the Remark 4.10: then we know that there are no η at the root position in $M_1' \Longrightarrow^* M_1''' \ {}^*\!\!\Longleftarrow M_1''$. By the induction hypothesis we get a term M_1''' that can be used to close the diagram $M_1' \Longleftarrow M_1 \Longrightarrow M_1''$ via $M_1' \Longrightarrow^* M_1''' \ {}^*\!\!\Longleftarrow M_1''$, and we can close our original diagram with

$$M' \equiv (M_1' M_2) \Longrightarrow^* (M_1''' M_2) \ {}^*\!\!\Longleftarrow (M_1'' M_2) \equiv M''$$

- $M_1' \Longleftarrow M_1 \Longrightarrow M_1''$ is one of the exceptional cases for η, hence M_1 is $\pi_1(\langle P, Q\rangle)$ for some terms P and Q. We can still close the original diagram as follows:

$$\begin{tikzcd}
(\pi_1(\langle P, Q\rangle))M_2 \arrow[r, Rightarrow, "\eta"] \arrow[dd, Rightarrow, "\pi"'] & (\pi_1(\langle \lambda x.Px, Q\rangle))M_2 \arrow[d, Rightarrow, "\pi"] \\
 & (\lambda x.Px)M_2 \arrow[d, Rightarrow, "\beta"] \\
PM_2 \arrow[r, phantom, "\equiv"] & PM_2
\end{tikzcd}$$

$$\begin{tikzcd}
(\pi_2(\langle P, Q\rangle))M_2 \arrow[r, Rightarrow, "\eta"] \arrow[dd, Rightarrow, "\pi"'] & (\pi_2(\langle P, \lambda x.Qx\rangle))M_2 \arrow[d, Rightarrow, "\pi"] \\
 & (\lambda x.Qx)M_2 \arrow[d, Rightarrow, "\beta"] \\
QM_2 \arrow[r, phantom, "\equiv"] & QM_2
\end{tikzcd}$$

– The reduction $M \equiv (M_1M_2) \Longrightarrow (M_1M_2'') \equiv M''$ comes from $M_2 \Longrightarrow M_2''$. We can close the diagram using the same original reductions,

$$M' \equiv (M_1'M_2) \Longrightarrow (M_1'M_2'') \Longleftarrow (M_1M_2'') \equiv M''$$

because we know that η is not applied to M_1 to get to M_1'.

— $M \equiv (M_1M_2) \Longrightarrow (M_1M_2') \equiv M'$ comes from $M_2 \Longrightarrow M_2'$. Then we have two cases:

– The reduction $M \equiv (M_1M_2) \Longrightarrow (M_1M_2'') \equiv M''$ comes from $M_2 \Longrightarrow M_2''$. By the induction hypothesis we get a term M_2''' that can be used to close the diagram $M_2' \Longleftarrow M_2 \Longrightarrow M_2''$ via $M_2' \Longrightarrow^* M_2''' \ {}^*\!\!\Longleftarrow M_2''$. Now $M' \equiv (M_1M_2') \Longrightarrow^* (M_1M_2''') \ {}^*\!\!\Longleftarrow (M_1M_2'') \equiv M''$ can be used to close our original diagram.

– The reduction $M \equiv (M_1M_2) \Longrightarrow (M_1''M_2) \equiv M''$ comes from $M_1 \Longrightarrow M_1''$. In this case, we know that η cannot be applied at the top to M_1 to get to M_1'' because M_1 is evaluated. So, we can close the diagram using the same original reductions as follows:

$$M' \equiv (M_1M_2') \Longrightarrow (M_1''M_2') \Longleftarrow (M_1''M_2) \equiv M''$$

— $M \equiv (\pi_i(M_1)) \Longrightarrow (\pi_i(M_1')) \equiv M'$ comes from $M_1 \Longrightarrow M_1'$ and $M \Longrightarrow (\pi_i(M_1'')) \equiv M''$ comes from $M_1 \Longrightarrow M_1''$. Then neither $M_1 \Longrightarrow M_1'$ nor $M_1 \Longrightarrow M_1''$ can use δ rules at the root of M because it is projected.
Now we have two cases:

– $M_1 \Longrightarrow M_1'$ and $M_1 \Longrightarrow M_1''$ are not the exceptional cases for δ of Remark 4.10. By the induction hypothesis there is an M_1''' s.t. $M_1' \Longrightarrow^* M_1''' \ {}^*\!\!\Longleftarrow M_1''$ without δ rules at the root, and we can close our diagram by $\pi_i(M_1') \Longrightarrow^* \pi_i(M_1''') \ {}^*\!\!\Longleftarrow \pi_i(M_1'')$.

– $M_1 \Longrightarrow M_1'$ and $M_1 \Longrightarrow M_1''$ is the exceptional case for δ, so $M_1 \equiv (\lambda x.P)Q$ for some terms P and Q. We can still close our original diagram as follows:

$$\begin{array}{ccc}
\pi_i((\lambda x.P)Q) & \stackrel{\delta}{\Longrightarrow} & \pi_i((\lambda x.\langle \pi_1(P), \pi_2(P)\rangle)Q) \\
 & & \Downarrow \beta \\
\beta \Big\downarrow & & \pi_i(\langle \pi_1(P[Q/x]), \pi_2(P[Q/x])\rangle) \\
 & & \Downarrow \pi \\
\pi_i(P[Q/x]) & \stackrel{\delta}{\Longrightarrow} & \pi_i(P[Q/x])
\end{array}$$

— $M \equiv \lambda x.M_1 \Longrightarrow \lambda x.M_1' \equiv M'$ comes from $M_1 \Longrightarrow M_1'$ and $M \equiv \lambda x.M_1 \Longrightarrow \lambda x.M_1'' \equiv M''$ comes from $M_1 \Longrightarrow M_1''$. By the induction hypothesis there is an M_1''' s.t. $M_1' \Longrightarrow^* M_1''' \ {}^*\!\!\Longleftarrow M_1''$ and then we can close our diagram by $\lambda x.M_1' \Longrightarrow^* \lambda x.M_1''' \ {}^*\!\!\Longleftarrow \lambda x.M_1''$.

— $M \equiv \langle M_1, M_2\rangle \Longrightarrow \langle M_1', M_2\rangle \equiv M'$ comes from $M_1 \Longrightarrow M_1'$. Now we have to consider two cases:

– The reduction $M \equiv \langle M_1, M_2\rangle \Longrightarrow \langle M_1, M_2''\rangle \equiv M''$ comes from a reduction $M_2 \Longrightarrow M_2''$. By the induction hypothesis there is a term M_1''' s.t. we can close the diagram $M_1' \Longleftarrow M_1 \Longrightarrow M_1''$ via $M_1' \Longrightarrow^* M_1''' \ {}^*\!\!\Longleftarrow M_1''$, and we can close our original diagram with

$$M' \equiv \langle M_1', M_2\rangle \Longrightarrow^* \langle M_1''', M_2\rangle \ {}^*\!\!\Longleftarrow \langle M_1'', M_2\rangle \equiv M''$$

– The reduction $M \equiv \langle M_1, M_2\rangle \Longrightarrow \langle M_1, M_2''\rangle \equiv M''$ comes from a reduction $M_2 \Longrightarrow M_2''$. We can close the diagram using the same original reductions,

$$M' \equiv \langle M_1, {}' M_2\rangle \Longrightarrow \langle M_1', M_2''\rangle \Longleftarrow \langle M_1, M_2''\rangle \equiv M''$$

— $M \equiv in_C^i(M_1) \Longrightarrow in_C^i(M_1') \equiv M'$ comes from $M_1 \Longrightarrow M_1'$ and $M \Longrightarrow in_C^i(M_1'') \equiv M''$ comes from $M_1 \Longrightarrow M_1''$. By the induction hypothesis there is an M_1''' s.t. $M_1' \Longrightarrow^* M_1''' \ {}^*\!\!\Longleftarrow M_1''$ and we can close our diagram by $in_C^i(M_1') \Longrightarrow^* in_C^i(M_1''') \ {}^*\!\!\Longleftarrow in_C^i(M_1'')$.

— $M \equiv rec\ x : A.M_1 \Longrightarrow rec\ x : A.M_1' \equiv M'$ comes from the reduction $M_1 \Longrightarrow M_1'$ and $M \equiv rec\ x : A.M_1 \Longrightarrow rec\ x : A.M_1'' \equiv M''$ comes from $M_1 \Longrightarrow M_1''$. Then we can find by the induction hypothesis an M_1''' s.t. $M_1' \Longrightarrow^* M_1''' \ {}^*\!\!\Longleftarrow M_1''$ and we can close our diagram by $rec\ x : A.M_1' \Longrightarrow^* rec\ x : A.M_1''' \ {}^*\!\!\Longleftarrow rec\ x : A.M_1''$.

— We are left to consider the case of $M \equiv Case(P, M_1, M_2)$.

– To avoid a mechanical repetition of similar proofs, notice that if the internal reduction to M' and M'' are performed on different subterms, then we can close the diagram by commuting the two reductions. We show just one case.

$$\begin{array}{ccc}
Case(P, M_1, M_2) & \stackrel{R2}{\Longrightarrow} & Case(P, M_1', M_2) \\
R1 \Downarrow & & \Downarrow R1 \\
Case(P', M_1, M_2) & \stackrel{R2}{\Longrightarrow} & Case(P', M_1', M_2)
\end{array}$$

– If the internal reduction to M' and M'' are performed on the same subterm Q, say for example $Q' \stackrel{R1}{\Longleftarrow} Q \stackrel{R2}{\Longrightarrow} Q''$, then there is a Q''', by the induction hypothesis, s.t. $Q' \Longrightarrow^* Q''' \ {}^*\!\!\Longleftarrow Q''$, and we can close the diagram by extending these last reductions to the $Case$ expression. Again, we detail just one case.

$$\begin{array}{ccc} Case(P, M_1, M_2) & \stackrel{R2}{\Longrightarrow} & Case(P'', M_1, M_2) \\ R1 \Big\Downarrow & & \Big\Downarrow \\ Case(P', M_1, M_2) & \Longrightarrow & Case(P''', M_1', M_2) \end{array}$$

□

5. Strong Normalization

We provide in this section the proof of strong normalization for our calculus. The key idea is to reduce strong normalization of the system with expansion rules to that of the system without expansion rules and for this, we show how the calculus without expansions can be used to simulate the calculus with expansions. We will use a fundamental property relating strong normalization of two systems:

Proposition 5.1. Let $\mathcal{R}_1$ and $\mathcal{R}_2$ be two reduction systems and $\mathcal{T}$ a translation from terms in $\mathcal{R}_1$ to terms in $\mathcal{R}_2$. If for every reduction $M_1 \stackrel{\mathcal{R}_1}{\Longrightarrow} M_2$ there is a non empty reduction sequence $P_1 \stackrel{\mathcal{R}_2}{\Longrightarrow}{}^+ P_2$ such that $\mathcal{T}(M_i) = P_i$, for $i = 1, 2$, then the strong normalization of $\mathcal{R}_2$ implies that of $\mathcal{R}_1$.

Proof. Suppose $\mathcal{R}_2$ is strongly normalizing and $\mathcal{R}_1$ is not. Then there is an infinite reduction sequence $M_1 \stackrel{\mathcal{R}_1}{\Longrightarrow} M_2 \stackrel{\mathcal{R}_1}{\Longrightarrow} \ldots$ and from this reduction we can construct an infinite reduction sequence $\mathcal{T}(M_1) \stackrel{\mathcal{R}_2}{\Longrightarrow}{}^+ \mathcal{T}(M_2) \stackrel{\mathcal{R}_2}{\Longrightarrow}{}^+ \ldots$ which leads to a contradiction. □

The goal is now to find a translation of terms mapping our calculus into itself such that for every possible reduction in the original system from a term M to another term N, there is a reduction sequence from the translation of M to the translation of N, that is *non empty* and *does not* contain any expansion. Then the previous proposition allows us to derive the strong normalization property for the full system from that of the system without expansion rules, which can be proved using standard techniques.

5.1. *Simulating Expansions without Expansions*

The first naïve idea that comes to the mind is to choose a translation such that expansion rules are completely impossible on a translated term. This essentially amounts to associate to a term M its η-δ normal form, so that translating a term corresponds then to executing all the possible expansions.

Unfortunately, this simple solution is not a good one: if M reduces to N via an expansion, then the translation of M and that of N are the same term, so to such a reduction step in the full system corresponds an *empty* reduction sequence in the translation, and this does not allow us to apply proposition 5.1.

This leads us to consider a more sophisticated translation that maps a term M to a term M° where expansions are not fully executed as above, but just *marked* in such a way that they can be executed during the simulation process, if necessary, by a rule that is not an expansion.

Let us see how to do this on a simple example: take a variable z of type $A_1 \times A_2$, where the A_i's are atomic types different from $\mathbf{T}$. By performing a δ expansion we obtain its normal form w.r.t. expansion rules: $\langle \pi_1(z), \pi_2(z) \rangle$. Instead of executing this reduction, we just *mark* it in the translation by applying to z an appropriate *expansor* term $\lambda x : A_1 \times A_2 . \langle \pi_1(x), \pi_2(x) \rangle$. As for $\langle \pi_1(z), \pi_2(z) \rangle$, it is in normal form w.r.t. expansions, so the translation does not modify it in any way. Now, we have the reduction sequence

$$z^\circ \equiv (\lambda x : A_1 \times A_2 . \langle \pi_1(x), \pi_2(x) \rangle) z \rightarrow_\beta \langle \pi_1(z), \pi_2(z) \rangle \equiv \langle \pi_1(z), \pi_2(z) \rangle^\circ$$

where the translation of z reduces to the translation of $\langle \pi_1(z), \pi_2(z)\rangle$, and the δ expansion from z to $\langle \pi_1(z), \pi_2(z)\rangle$ is simulated in the translation by a β-rule. Clearly, in a generic term M there are many positions where an expansion can be performed, so the translation will have to take into account the *structure* of M and insert the appropriate expansors at all these positions[‖].

Anyway, expansors must be carefully defined to correctly represent not only the expansion step arising from a redex already present in M, but also all the expansion sequences that such step can create: if in the previous example the type A_1 is taken to be an arrow type and the type A_2 a product type, then the term $\pi_1(z)$ can be further η-expanded and the term $\pi_2(z)$ can be expanded by a δ-rule, and the expansor $\lambda x : A_1 \times A_2.\langle \pi_1(x), \pi_2(x)\rangle$ cannot simulate these further possible reductions. This can only be done by storing in the expansor terms all the information on possible future expansions, that is fully contained in the *type* of the term we are marking.

Definition 5.2. (Translation) To every type C we associate a term, called the *expansor of type* C and denoted Δ_C, defined by induction as follows:

$$\begin{array}{ll} \Delta_{A\to B} = & \lambda x : A \to B.\lambda z : A.\Delta_B(x(\Delta_A z)) \\ \Delta_{A\times B} = & \lambda x : A \times B.\langle \Delta_A(\pi_1(x)), \Delta_B(\pi_2(x))\rangle \\ \Delta_A & is\ empty, \text{ in any other case} \end{array}$$

We then define a translation M° for a term $M : A$ as follows:

$$M^\circ = \begin{cases} M^{\circ\circ} & \text{if } M \text{ is a } \lambda\text{-abstraction or a pair} \\ \Delta_A^k M^{\circ\circ} \quad \text{for any } k > 0 & \text{otherwise} \end{cases}$$

where $\Delta_A^k M$ denotes the term $\underbrace{(\Delta_A \ldots (\Delta_A}_{k \quad times} M) \ldots)$ and $M^{\circ\circ}$ is defined by induction as:

$$\begin{array}{llll} x^{\circ\circ} & = x & (\lambda x : B.M)^{\circ\circ} & = \lambda x : B.M^\circ \\ *^{\circ\circ} & = * & (rec\, y : A.M)^{i\,\circ\circ} & = (rec\, y : A.M^\circ)^i \\ \langle M, N\rangle^{\circ\circ} & = \langle M^\circ, N^\circ\rangle & Case(R, M, N)^{\circ\circ} & = Case(R^\circ, M^\circ, N^\circ) \\ (MN)^{\circ\circ} & = (M^{\circ\circ} N^\circ) & \pi_i(M)^{\circ\circ} & = \pi_i(M^{\circ\circ}) \\ in_C^i(M)^{\circ\circ} & = in_C^i(M^\circ) & & \end{array}$$

This corresponds exactly to the marking procedure described before, but for a little detail: in the translation we allow *any* number of markers to be used (the integer k can be any positive number), and not just one as seemed to suffice for the examples above.

The need for this additional twist in the definition is best understood with an example. Consider two atomic types A and B and the term $(\lambda x : A \times B.x)z$: if k is fixed to be one (*i.e.* we allow only one expansor as marker) then its translation $((\lambda x : A \times B.x)z)^\circ$ is $\Delta_{A\times B}((\lambda x : A \times B.\Delta_{A\times B} x)\Delta_{A\times B} z)$. Now $(\lambda x : A \times B.x)z \xrightarrow{\beta} z$, so we have to verify that $((\lambda x : A \times B.x)z)^\circ$ reduces to z° in at least one step. We have:

$$\Delta_{A\times B}((\lambda x : A \times B.\Delta_{A\times B} x)\Delta_{A\times B} z) \Longrightarrow \Delta_{A\times B}\Delta_{A\times B}\Delta_{A\times B} z$$

However, even if both $\Delta^3_{A\times B} z$ and $\Delta_{A\times B} z$ reduce to the same term $\langle \pi_1(z), \pi_2(z)\rangle$, it is not true that $\Delta^3_{A\times B} z \Longrightarrow^* \Delta_{A\times B} z$. Anyway, if we admit $\Delta^3_{A\times B} z$ as a possible translation of z we will have the desired property relating reductions and translations. Hence, to be precise, our method associates to each term not

[‖] Notice that we cannot insert expansors in influential positions: if a term M is expanded, say to $\langle \pi_1(M), \pi_2(M)\rangle$, then its root becomes an influential position, and we cannot insure that the translation of M reduces to a translation of $\langle \pi_1(M), \pi_2(M)\rangle$: expansors get used, and after some reduction steps we end up with a naked pair not preceded by an expansor.

just one translation, but a whole family of possible translations, all with the same structure, but with different numbers of expansors used as markers.

What is important for our proof is that given a reduction sequence $M_1 \Longrightarrow M_2 \ldots \Longrightarrow M_n$ in the full calculus, then no matter which possible translation M_1° we choose for M_1, the reductions used in the simulation process all go through possible translations M_i° of the M_i.

Translations preserve types and leave unchanged terms where expansions are not possible.

Lemma 5.3. If $\Gamma \vdash M : A$, then $\Gamma \vdash (\Delta_A M) : A$.

Proof. By induction on the structure of A.

— If A is neither a functional, nor a product type, then Δ_A is empty and the property trivially holds.
— $A \equiv B \to C$. Since $\Gamma, x : B \to C, z : B \vdash z : B$, we have by the induction hypothesis $\Gamma, x : B \to C, z : B \vdash (\Delta_B z) : B$

$$\frac{\Gamma, x : B \to C, z : B \vdash x : B \to C \qquad \Gamma, x : B \to C, z : B \vdash (\Delta_B z) : B}{\Gamma, x : B \to C, z : B \vdash (x(\Delta_B z)) : C}$$

Again by the induction hypothesis $\Gamma, x : B \to C, z : B \vdash \Delta_C(x(\Delta_B z)) : C$ and thus:

$$\frac{\dfrac{\dfrac{\Gamma, x : B \to C, z : B \vdash \Delta_C(x(\Delta_B z)) : C}{\Gamma, x : B \to C \vdash \lambda z : B.\Delta_C(x(\Delta_B z)) : B \to C}}{\Gamma \vdash \lambda x : B \to C.\lambda z : B.\Delta_C(x(\Delta_B z)) : (B \to C) \to (B \to C)} \qquad \Gamma \vdash M : B \to C}{\Gamma \vdash (\Delta_{B\to C} M) : B \to C}$$

— $A \equiv B \times C$. Since $\Gamma, x : B \times C \vdash x : B \times C$, then $\Gamma, x : B \times C \vdash \pi_1(x) : B$ and $\Gamma, x : B \times C \vdash \pi_2(x) : C$. By the induction hypothesis $\Gamma, x : B \times C \vdash \Delta_B \pi_1(x) : B$ and $\Gamma, x : B \times C \vdash \Delta_C \pi_1(x) : C$.

$$\frac{\dfrac{\dfrac{\Gamma, x : B \times C \vdash \Delta_B \pi_1(x) : B \qquad \Gamma, x : B \times C \vdash \Delta_C \pi_2(x) : C}{\Gamma, x : B \times C \vdash \langle \Delta_B \pi_1(x), \Delta_C \pi_2(x) \rangle : B \times C}}{\Gamma \vdash \lambda x : B \times C.\langle \Delta_B \pi_1(x), \Delta_C \pi_2(x) \rangle : (B \times C) \to (B \times C)} \qquad \Gamma \vdash M : B \times C}{\Gamma \vdash \Delta_{B\times C} M : B \times C}$$

□

Corollary 5.4. If $\Gamma \vdash M : A$, then $\Gamma \vdash \Delta_A^k M : A$, for any $k \geq 0$.

Lemma 5.5. (Type Preservation) If $\Gamma \vdash M : A$, then $\Gamma \vdash M^\circ : A$ and $\Gamma \vdash M^{\circ\circ} : A$.

Proof. By induction on the structure of M, using corollary 5.4. the induction hypothesis

□

A term M is in *quasi-normal form* if only expansion rules at the *root position* are applicable to it and M is in *normal form* if no rule is applicable to it. So, every normal form is in quasi-normal form, while the converse does not necessarily hold.

Lemma 5.6.

1 If M is in normal form, then $M^\circ = M$

2 If M is in quasi-normal form, then $M^{\circ\circ} = M$

Proof. By induction on the structure of M.

— $M \equiv *$.

1 $*^{\circ} = *$.

2 The property vacuously holds because $*$ is a normal form.

— $M \equiv x$.

1 Since x is in normal form, it has neither a functional, nor a product, nor the $\mathbf{T}$ type and then Δ_A is empty, where A is the type of x. Then $x^{\circ} = x$.

2 $x^{\circ\circ} = x$ by definition.

— $M \equiv \lambda x : A.P$.

1 Since M is in normal form, P is also in normal form and by the induction hypothesis $P^{\circ} = P$. We have $(\lambda x : A.P)^{\circ} = \lambda x : A.P^{\circ} = \lambda x : A.P$.

2 If $\lambda x : A.P$ is in quasi-normal form, it is also in normal form because we cannot apply an expansion rule to a $lambda$-term. By the previous paragraph $(\lambda x : A.P)^{\circ\circ} = \lambda x : A.P$.

— $M \equiv \langle P, Q \rangle$.

1 Since M is in normal form, P and Q are also in normal form and by the induction hypothesis $P^{\circ} = P$ and $Q^{\circ} = Q$. We have $\langle P, Q \rangle^{\circ} = \langle P^{\circ}, Q^{\circ} \rangle = \langle P, Q \rangle$.

2 If $\langle P, Q \rangle$ is in quasi-normal form, it is also in normal form because we cannot apply an expansion rule to a pair. By the previous paragraph $\langle P, Q \rangle^{\circ\circ} = \langle P, Q \rangle$.

— $M \equiv (rec\ y : A.P)^i$.

– If $i = 0$, then

1 Since M is in normal form, P is also in normal form and by the induction hypothesis $P^{\circ} = P$. On the other hand, M has neither a functional, nor a product, nor the $\mathbf{T}$ type and then Δ_A is empty, where A is the type of M. We have $((rec\ y : A.P)^0)^{\circ} = \Delta_A^k (rec\ y : A.P^{\circ})^0 = (rec\ y : A.P)^0$.

2 Since M is in quasi normal form, P is in normal form and by the induction hypothesis $P^{\circ} = P$. Then $((rec\ y : A.P)^0)^{\circ} = (rec\ y : A.P^{\circ})^0 = (rec\ y : A.P)^0$.

– If $i > 0$, then

1 The property vacuously holds because $(rec\ y : A.P)^i$ is not in normal form.

2 The property vacuously holds because $(rec\ y : A.P)^i$ is not in quasi-normal form.

— $M \equiv (PQ)$.

1 Suppose A is the type of M. Since M is in normal form, A is neither a functional, nor a product, nor the $\mathbf{T}$ type and so Δ_A is empty. On the other hand P is in quasi-normal form and Q is in normal form, so by the induction hypothesis $P^{\circ\circ} = P$ and $Q^{\circ} = Q$. We have $(PQ)^{\circ} = \Delta_A^k (P^{\circ\circ} Q^{\circ}) = (PQ)$.

2 Since M is in quasi-normal form, P is in quasi-normal form and Q is in normal form and by the induction hypothesis $P^{\circ\circ} = P$ and $Q^{\circ} = Q$. We have $(PQ)^{\circ\circ} = (P^{\circ\circ} Q^{\circ}) = (PQ)$.

— $M \equiv Case(P, R, N)$.

1 Suppose A is the type of M. Since M is in normal form, A is neither a functional, nor a product, nor the $\mathbf{T}$ type and so Δ_A is empty. On the other hand P, R and N are in normal form and by the induction

hypothesis $P^\circ = P$ and $R^\circ = R$ and $N^\circ = R$. We have $Case(P,R,N)^\circ = \Delta_A^k\ Case(P^\circ, R^\circ, N^\circ) = Case(P,R,N)$.

2 Since M is in quasi-normal form, P, R and N are in normal form and by the induction hypothesis $P^\circ = P$ and $R^\circ = R$ and $N^\circ = R$. We have $Case(P,R,N)^{\circ\circ} = Case(P^\circ, R^\circ, N^\circ) = Case(P,R,N)$.

— $M \equiv \pi_i(P)$, for $i = 1, 2$.

1 Suppose A is the type of M. Since M is in normal form, A is neither a functional, nor a product, nor the $\mathbf{T}$ type and so Δ_A is empty. On the other hand P is in quasi-normal form and by the induction hypothesis $P^{\circ\circ} = P$. We have $\pi_i(P)^\circ = \Delta_A^k\ \pi_i(P^{\circ\circ}) = \pi_i(P)$.

2 Since M is in quasi-normal form, P is also in quasi-normal form and by the induction hypothesis $P^{\circ\circ} = P$. We have $\pi_i(P)^{\circ\circ} = \pi_i(P^{\circ\circ}) = \pi_i(P)$.

— $M \equiv in_C^i(P)$, for $i = 1, 2$.

1 Since M is in normal form, P is also in normal form and by the induction hypothesis $P^\circ = P$. We have $in_C^i(P)^\circ = in_C^i(P^\circ) = in_C^i(P)$.

2 $in_C^i(P)$ in quasi-normal form implies $in_C^i(P)$ in normal form, and the property holds by the previous paragraph.

□

The next step is to prove that we can apply proposition 5.1 to our system, *i.e,* for every one step reduction from M to N in the full system, there is a non empty reduction sequence in the system without expansions from any translation of M to a translation of N.

This lemma characterizes the reductions from a term $\Delta^k_{A\to B}M$ or $\Delta^k_{A\times B}M$ and is quite essential in all the properties shown in this section.

Lemma 5.7. For any $k > 0$

$$\Delta^k_{A\to B}M \Longrightarrow^+ \lambda w : A.\Delta^k_B(M(\Delta^k_A w)) \quad \text{and} \quad \Delta^k_{A\times B}M \Longrightarrow^+ \langle \Delta^k_A \pi_1(M), \Delta^k_B \pi_2(M)\rangle$$

and the reduction sequences contain no expansion steps.

Proof. By induction on k.
If $k = 1$, then

$$\Delta_{A\to B}M \equiv (\lambda x : A \to B.\lambda w : A.\Delta_B(x(\Delta_A w)))M \xrightarrow{\beta} \lambda w : A.\Delta_B(M(\Delta_A w))$$

$$\Delta_{A\times B}M \equiv (\lambda x : A \times B.\langle \Delta_A \pi_1(x), \Delta_B \pi_2(x)\rangle)M \xrightarrow{\beta} \langle \Delta_A \pi_1(M), \Delta_B \pi_2(M)\rangle$$

Only the β-rule is used in this reduction.
If $k > 1$, then

$$\Delta^{k+1}_{A\to B} M = \Delta^k_{A\to B}\Delta_{A\to B} M$$

$$\Downarrow_\beta$$

$$\Delta^k_{A\to B}\lambda w : A.\Delta_B(M(\Delta_A w))$$

$\Downarrow_+$ by the induction hypothesis (and without expansions steps)

$$\lambda w : A.\Delta^k_B(\lambda w : A.\Delta_B(M(\Delta_A w))(\Delta^k_A w))$$

$$\Downarrow_\beta$$

$$\lambda w : A.\Delta^k_B(\Delta_B(M(\Delta_A\Delta^k_A w))) = \lambda w : A.\Delta^{k+1}_B(M(\Delta^{k+1}_A w))$$

$$\Delta^{k+1}_{A\times B} M = \Delta^k_{A\times B}(\Delta_{A\times B} M)$$

$$\Downarrow_\beta$$

$$\Delta^k_{A\times B}\langle\Delta_A\pi_1(M), \Delta_B\pi_2(M)\rangle$$

$\Downarrow_+$ by the induction hypothesis (and without expansion steps)

$$\langle\Delta^k_A\pi_1(\langle\Delta_A\pi_1(M), \Delta_B\pi_2(M)\rangle), \Delta^k_B\pi_2(\langle\Delta_A\pi_1(M), \Delta_B\pi_2(M)\rangle)\rangle$$

$$\Downarrow_{\pi_1,\pi_2}$$

$$\langle\Delta^k_A\Delta_A\pi_1(M), \Delta^k_B\Delta_B\pi_2(M)\rangle = \langle\Delta^{k+1}_A\pi_1(M), \Delta^{k+1}_B\pi_2(M)\rangle$$

□

We use $N^\otimes$ to denote either N° or $N^{\circ\circ}$. In particular, $\overline{N^\otimes}$ will stand for a sequence of mixed N_i°'s and $N_i^{\circ\circ}$'s.

Lemma 5.8. If $\Gamma \vdash M : A$, then for any substitution $[\overline{z^\otimes}/\overline{x}]$ we have

1 $\exists k \geq 0,\ M^{\circ\circ}[\overline{z^\otimes}/\overline{x}] \Longrightarrow^* \ \Delta^k_A(M[\overline{z}/\overline{x}])^{\circ\circ}$
2 $\forall k \geq 0,\ \Delta^k_A M^\circ[\overline{z^\otimes}/\overline{x}] \Longrightarrow^* \ (M[\overline{z}/\overline{x}])^\circ$
and no expansions are performed in these reduction sequences.

Proof. We show the two properties by induction on the structure of M. More precisely, for the first statement we analyze each case, while for the second one it is enough to analyze those expressions M such that $M^\circ = M^{\circ\circ}$. Indeed, once we have already shown the first statement, the second can be easily shown in the following way for the expressions M such that $M^\circ = \Delta^h_A M^{\circ\circ}$ (for $h > 0$):

$\Delta_A^k M^\circ[\overline{z^\otimes}/\overline{x}] =$
$\Delta_A^k \Delta_A^h M^{\circ\circ}[\overline{z^\otimes}/\overline{x}] \Longrightarrow^*$ (by the first statement)
$\Delta_A^{k+h} \Delta_A^m M[\overline{z}/\overline{x}]^{\circ\circ} =$ (as $h > 0$)
$M[\overline{z}/\overline{x}]^\circ$

Every reduction built in the following proof contains no expansion steps, as it is constructed from one-step reductions that are not expansions or from reductions obtained by the induction hypothesis (and thus without expansions) or from reductions obtained by lemma 5.7 (again without expansions). This remark will allow us to conclude that the reductions in the statements of the lemma contain no expansion.
Now, let us analyze the first statement and the interesting cases of the second.

— $M \equiv *$. Since $*$ is of type $\mathbf{T}$, $\Delta_{\mathbf{T}}$ is empty. We have

$$*^{\circ\circ}[\overline{z^\otimes}/\overline{x}] = *[\overline{z^\otimes}/\overline{x}] = * = *^{\circ\circ} = \Delta^0_{\mathbf{T}} *^{\circ\circ} = \Delta^0_{\mathbf{T}}(*[\overline{z}/\overline{x}])^{\circ\circ}$$

— $M \equiv x_i \in \overline{x}$. There are two cases to consider: either $z^\otimes = z^\circ$ or $z^\otimes = z^{\circ\circ}$.

- $x_i^{\circ\circ}[\overline{z^\otimes}/\overline{x}] = x_i[\overline{z^\otimes}/\overline{x}] = z_i^\circ = \Delta_A^m z_i^{\circ\circ} = \Delta_A^m(x_i[\overline{z}/\overline{x}])^{\circ\circ}$.
- $x_i^{\circ\circ}[\overline{z^\otimes}/\overline{x}] = x_i[\overline{z^\otimes}/\overline{x}] = z_i^{\circ\circ} = (x_i[\overline{z}/\overline{x}])^{\circ\circ} = \Delta_A^0(x_i[\overline{z}/\overline{x}])^{\circ\circ}$.

— $M \equiv y \notin \overline{x}$. We have $y^{\circ\circ}[\overline{z^\otimes}/\overline{x}] = y[\overline{z^\otimes}/\overline{x}] = y = \Delta_A^0 y^{\circ\circ}$.

— $M \equiv (PQ)$. We have $(PQ)^{\circ\circ}[\overline{z^\otimes}/\overline{x}] = (P^{\circ\circ}Q^\circ)[\overline{z^\otimes}/\overline{x}] = P^{\circ\circ}[\overline{z^\otimes}/\overline{x}]Q^\circ[\overline{z^\otimes}/\overline{x}]$.
By the induction hypothesis $P^{\circ\circ}[\overline{z^\otimes}/\overline{x}] \Longrightarrow^* \Delta^h_{B\to A}P[\overline{z}/\overline{x}]^{\circ\circ}$

- If $h = 0$, then

$P^{\circ\circ}[\overline{z^\otimes}/\overline{x}]Q^\circ[\overline{z^\otimes}/\overline{x}]$

$\Downarrow_*$ by the induction hypothesis

$P[\overline{z}/\overline{x}]^{\circ\circ}Q^\circ[\overline{z^\otimes}/\overline{x}]$

$\Downarrow_*$ by the induction hypothesis

$P[\overline{z}/\overline{x}]^{\circ\circ}Q[\overline{z}/\overline{x}]^\circ = (P[\overline{z}/\overline{x}]Q[\overline{z}/\overline{x}])^{\circ\circ} = ((PQ)[\overline{z}/\overline{x}])^{\circ\circ} = \Delta_A^0(PQ[\overline{z}/\overline{x}])^{\circ\circ}$

- If $h > 0$, then:

$\Delta^h_{B\to A} P[\overline{z}/\overline{x}]^{\circ\circ} Q^\circ[\overline{z^\otimes}/\overline{x}]$

$\Downarrow_+$ by lemma 5.7

$(\lambda w : B.\Delta^h_A(P[\overline{z}/\overline{x}]^{\circ\circ}(\Delta^h_B w)))Q^\circ[\overline{z^\otimes}/\overline{x}]$

$\Downarrow_\beta$

$\Delta^h_A(P[\overline{z}/\overline{x}]^{\circ\circ}(\Delta^h_B(Q^\circ[\overline{z^\otimes}/\overline{x}])))$

$\Downarrow_*$ by the induction hypothesis

$\Delta^h_A(P[\overline{z}/\overline{x}]^{\circ\circ} Q[\overline{z}/\overline{x}]^\circ) = \Delta^h_A(P[\overline{z}/\overline{x}]Q[\overline{z}/\overline{x}])^{\circ\circ} = \Delta^h_A(PQ[\overline{z}/\overline{x}])^{\circ\circ}$

— $M \equiv \lambda y : B.P$.

1 for the first statement,

$(\lambda y : B.P)^{\circ\circ}[\overline{z^\otimes}/\overline{x}] = (\lambda y : B.P^\circ)[\overline{z^\otimes}/\overline{x}]$
$=$
$\lambda y : B.P^\circ[\overline{z^\otimes}/\overline{x}]$

$\Downarrow_*$ by the induction hypothesis

$\lambda y : B.P[\overline{z}/\overline{x}]^\circ = (\lambda y : B.P[\overline{z}/\overline{x}])^{\circ\circ} = \Delta^0_{B\to C}((\lambda y : B.P)[\overline{z}/\overline{x}])^{\circ\circ}$

2 for the second statement,

$$\begin{array}{l}
\Delta^k_{B\to C}(\lambda y : B.P)^\circ[\overline{z^\otimes}/\overline{x}] = \Delta^k_{B\to C}(\lambda y : B.P^\circ)[\overline{z^\otimes}/\overline{x}] \\
= \\
\Delta^k_{B\to C}\lambda y : B.P^\circ[\overline{z^\otimes}/\overline{x}] \\
\\
\Downarrow_+ \;\text{ by lemma 5.7} \\
\\
\lambda w : B.\Delta^k_B((\lambda y : B.P^\circ[\overline{z^\otimes}/\overline{x}])(\Delta^k_C w)) \\
\\
\Downarrow_\beta \\
\\
\lambda w : B.\Delta^k_B P^\circ[\overline{z^\otimes}/\overline{x}][w^\circ/y] \\
\\
\Downarrow_* \;\text{ by the induction hypothesis} \\
\\
\lambda w : B.(P[\overline{z}/\overline{x}][w/y])^\circ \\
= \\
(\lambda w : B.P[\overline{z}/\overline{x}][w/y])^\circ \\
=_\alpha \\
(\lambda y : B.P[\overline{z}/\overline{x}])^\circ = ((\lambda y : B.P)[\overline{z}/\overline{x}])^\circ
\end{array}$$

— $M \equiv \pi_i(P)$, for $i = 1, 2$.

$$\begin{array}{l}
\pi_i(P)^{\circ\circ}[\overline{z^\otimes}/\overline{x}] = \pi_i(P^{\circ\circ})[\overline{z^\otimes}/\overline{x}] \\
= \\
\pi_i(P^{\circ\circ}[\overline{z^\otimes}/\overline{x}]) \\
\\
\Downarrow_* \;\text{ by the induction hypothesis} \\
\\
\pi_i(\Delta^h_{A_1\times A_2}(P[\overline{z}/\overline{x}]^{\circ\circ})
\end{array}$$

– If $h = 0$, then

$$\pi_i(P[\overline{z}/\overline{x}]^{\circ\circ}) = \pi_i(P[\overline{z}/\overline{x}])^{\circ\circ} = \Delta^0_{A_i}\pi_i(P[\overline{z}/\overline{x}])^{\circ\circ} = \Delta^0_{A_i}(\pi_i(P)[\overline{z}/\overline{x}])^{\circ\circ}$$

– If $h > 0$, then

$$\begin{array}{l}
\pi_i(\Delta^h_{A_1\times A_2}(P[\overline{z}/\overline{x}])^{\circ\circ}) \\
\\
\Downarrow_+ \;\text{ by lemma 5.7} \\
\\
\pi_i(\langle \Delta^h_{A_1}\pi_1((P[\overline{z}/\overline{x}])^{\circ\circ}), \Delta^h_{A_2}\pi_2((P[\overline{z}/\overline{x}])^{\circ\circ})\rangle) \\
\\
\Downarrow_{\pi_i} \\
\\
\Delta^h_{A_i}\pi_i((P[\overline{z}/\overline{x}])^{\circ\circ}) = \Delta^h_{A_i}\pi_i(P[\overline{z}/\overline{x}])^{\circ\circ} = \Delta^h_{A_i}(\pi_i(P)[\overline{z}/\overline{x}])^{\circ\circ}
\end{array}$$

— $M \equiv in^i_C(P)$, for $i = 1, 2$ and then Δ_C is empty.

$$in^i_C(P)^{\circ\circ}[\overline{z^{\otimes}}/\overline{x}] = in^i_C(P^{\circ})[\overline{z^{\otimes}}/\overline{x}]$$
$$=$$
$$in^i_C(P^{\circ}[\overline{z^{\otimes}}/\overline{x}])$$

$\Downarrow_*$ by the induction hypothesis

$$in^i_C(P[\overline{z}/\overline{x}]^{\circ}) = \Delta^0_A in^i_C(P[\overline{z}/\overline{x}])^{\circ\circ} = \Delta^0_A(in^i_C(P)[\overline{z}/\overline{x}])^{\circ\circ}$$

— $M \equiv \langle P, Q\rangle$.

1 for the first statement,

$$\langle P, Q\rangle^{\circ\circ}[\overline{z^{\otimes}}/\overline{x}] = \langle P^{\circ}, Q^{\circ}\rangle[\overline{z^{\otimes}}/\overline{x}]$$
$$=$$
$$\langle P^{\circ}[\overline{z^{\otimes}}/\overline{x}], Q^{\circ}[\overline{z^{\otimes}}/\overline{x}]\rangle$$

$\Downarrow_*$ by the induction hypothesis

$$\langle P[\overline{z}/\overline{x}]^{\circ}, Q[\overline{z}/\overline{x}]^{\circ}\rangle$$
$$=$$
$$\langle P[\overline{z}/\overline{x}], Q[\overline{z}/\overline{x}]\rangle^{\circ\circ} = \Delta^0_{A\times B}\langle P[\overline{z}/\overline{x}], Q[\overline{z}/\overline{x}]\rangle^{\circ\circ} = \Delta^0_{A\times B}(\langle P, Q\rangle[\overline{z}/\overline{x}])^{\circ\circ}$$

2 for the second statement,

$$\Delta^k_{A\times B}\langle P, Q\rangle^{\circ}[\overline{z^{\otimes}}/\overline{x}] = \Delta^k_{A\times B}\langle P^{\circ}, Q^{\circ}\rangle[\overline{z^{\otimes}}/\overline{x}] = \Delta^k_{A\times B}\langle P^{\circ}[\overline{z^{\otimes}}/\overline{x}], Q^{\circ}[\overline{z^{\otimes}}/\overline{x}]\rangle$$

– If $k = 0$, then

$$\langle P^{\circ}[\overline{z^{\otimes}}/\overline{x}], Q^{\circ}[\overline{z^{\otimes}}/\overline{x}]\rangle$$

$\Downarrow_*$ by the induction hypothesis

$$\langle P[\overline{z}/\overline{x}]^{\circ}, Q[\overline{z}/\overline{x}]^{\circ}\rangle$$
$$=$$
$$\langle P[\overline{z}/\overline{x}], Q[\overline{z}/\overline{x}]\rangle^{\circ} = (\langle P, Q\rangle[\overline{z}/\overline{x}])^{\circ}$$

– If $k > 0$, then

$$\Delta^k_{A\times B}\langle P^\circ[\overline{z^\otimes}/\overline{x}], Q^\circ[\overline{z^\otimes}/\overline{x}]\rangle$$

$\Downarrow_+$ by lemma 5.7

$$\langle \Delta^k_A \pi_1(P^\circ[\overline{z^\otimes}/\overline{x}]Q^\circ[\overline{z^\otimes}/\overline{x}]), \Delta^k_B \pi_2(\langle P^\circ[\overline{z^\otimes}/\overline{x}], Q^\circ[\overline{z^\otimes}/\overline{x}]\rangle)\rangle$$

$\Downarrow_* \; \pi_1, \pi_2$

$$\langle \Delta^k_A P^\circ[\overline{z^\otimes}/\overline{x}], \Delta^k_B Q^\circ[\overline{z^\otimes}/\overline{x}]\rangle$$

$\Downarrow_*$ by the induction hypothesis

$$\langle P[\overline{z}/\overline{x}]^\circ, Q[\overline{z}/\overline{x}]^\circ\rangle = \langle P[\overline{z}/\overline{x}], Q[\overline{z}/\overline{x}]\rangle^\circ = (\langle P, Q\rangle[\overline{z}/\overline{x}])^\circ$$

— $M \equiv (rec\ y : A.P)^i$.

$$((rec\ y : A.P)^i)^{\circ\circ}[\overline{z^\otimes}/\overline{x}]$$
$$=$$
$$(rec\ y : A.P^\circ)^i[\overline{z^\otimes}/\overline{x}]$$

$\Downarrow_*$ by the induction hypothesis

$$(rec\ y : A.(P[\overline{z}/\overline{x}])^\circ)^i = \Delta^0_A((rec\ y : A.P[\overline{z}/\overline{x}])^i)^{\circ\circ} = \Delta^0_A((rec\ y : A.P)^i[\overline{z}/\overline{x}])^{\circ\circ}$$

— $M \equiv Case(P, Q, R)$.

$$Case(P, Q, R)^{\circ\circ}[\overline{z^\otimes}/\overline{x}] = Case(P^\circ, Q^\circ, R^\circ)[\overline{z^\otimes}/\overline{x}]$$
$$=$$
$$Case(P^\circ[\overline{z^\otimes}/\overline{x}], Q^\circ[\overline{z^\otimes}/\overline{x}], R^\circ[\overline{z^\otimes}/\overline{x}])$$

$\Downarrow_*$ by the induction hypothesis

$$Case(P[\overline{z}/\overline{x}]^\circ, Q[\overline{z}/\overline{x}]^\circ, R[\overline{z}/\overline{x}]^\circ)$$
$$=$$
$$Case(P[\overline{z}/\overline{x}], Q[\overline{z}/\overline{x}], R[\overline{z}/\overline{x}])^{\circ\circ}$$
$$=$$
$$(Case(P, Q, R)[\overline{z}/\overline{x}])^{\circ\circ}$$
$$=$$
$$\Delta^0_A(Case(P, Q, R)[\overline{z}/\overline{x}])^{\circ\circ}$$

□

Corollary 5.9. If $\Gamma \vdash M : A$, then $\forall k \geq 0,\ \Delta^k_A M^\circ \Longrightarrow^* \ M^\circ$ and no expansions are performed in the reduction sequences.

The following property is essential to show that every time we perform a β-reduction on a term M in the original system, any translation of M reduces to a translation of the term we have obtained via $\rightarrow_\beta$ from M.

Take for example the reduction $(\lambda x : A.M)N \rightarrow_\beta M[N/x]$. We know that $((\lambda x : A.M)N)^\circ = \Delta_A^k((\lambda x : A.M^\circ)N^\circ)$ and we want to show that there is a *non empty* reduction sequence leading to $M[N/x]^\circ$. Since $\Delta_A^k((\lambda x : A.M^\circ)N^\circ) \rightarrow_\beta \Delta_A^k M^\circ[N^\circ/x]$, we have now to check that the term $(M[N/x])^\circ$ can be reached. We state the property as follows:

Lemma 5.10. If $\Gamma \vdash M : A$, then

1 $\exists k \geq 0,\ M^{\circ\circ}[\overline{N^\otimes}/\overline{x}] \Longrightarrow^* \ \Delta_A^k(M[\overline{N}/\overline{x}])^{\circ\circ}$
2 $\forall k \geq 0,\ \Delta_A^k M^\circ[\overline{N^\otimes}/\overline{x}] \Longrightarrow^* \ (M[\overline{N}/\overline{x}])^\circ$
and no expansions are performed in the reduction sequences.

Proof. We show the two properties by induction on the structure of M. More precisely, for the first statement one analyzes each case, while for the second one it is enough to analyze those expressions M such that $M^\circ = M^{\circ\circ}$. Indeed, once we have already shown the first statement, the second can be easily shown in the following way for the expressions M such that $M^\circ = \Delta_A^h M^{\circ\circ}$ (for $h > 0$):

$$\begin{array}{l}
\Delta_A^k M^\circ[\overline{N^\otimes}/\overline{x}] = \\
\Delta_A^k \Delta_A^h M^{\circ\circ}[\overline{N^\otimes}/\overline{x}] \Longrightarrow^* \ \text{(by the first statement)} \\
\Delta_A^{k+h} \Delta_A^m M[\overline{N}/\overline{x}]^{\circ\circ} \Longrightarrow^* \ \text{(either by definition of } M[\overline{N}/\overline{x}]^\circ \text{ or by corollary 5.9)} \\
M[\overline{N}/\overline{x}]^\circ
\end{array}$$

Now, the analysis of all the cases involved proceeds exactly as in lemma 5.8, except for the case $M \equiv x_i$, where to prove the second statement we need to proceed as follows:

$$\Delta_A^k x_i^\circ[\overline{N^\otimes}/\overline{x}] = \Delta_A^k(\Delta_A^m x_i)[\overline{N^\otimes}/\overline{x}] = \Delta_A^{k+m} N_i^\otimes$$

Now, if $N_i^\otimes$ is N_i°, by corollary 5.9, $\Delta_A^{k+m} N_i^\circ \Longrightarrow^* \ N_i^\circ = (x_i[\overline{N}/\overline{x}])^\circ$ as required.

If instead $N_i^\otimes$ is $N_i^{\circ\circ}$, we have two cases:

— $N_i^\circ = N_i^{\circ\circ}$, then by corollary 5.9 $\Delta_A^{k+m} N_i^\circ \Longrightarrow^* \ N_i^\circ = (x_i[\overline{N}/\overline{x}])^\circ$.
— otherwise, $\Delta_A^{k+m} N_i^{\circ\circ} = N_i^\circ = (x_i[\overline{N}/\overline{x}])^\circ$.

Notice that there are no expansions in these reduction sequences, because of corollary 5.9.

□

Lemma 5.11. If $M \overset{\eta,\delta,Top}{\longrightarrow} N$, then $M^\circ \overset{\neg\delta,\neg\eta}{\Longrightarrow}{}^+ N^\circ$.

Proof.

— If M is of type $A \times B$ and $M \overset{\delta}{\longrightarrow} \langle \pi_1(M), \pi_2(M) \rangle$
We know that $\exists k > 0$ such that $M^\circ = \Delta_{A\times B}^k M^{\circ\circ}$. By corollary 5.7

$$\Delta_{A\times B}^k M^{\circ\circ} \Longrightarrow^+ \langle \Delta_A^k \pi_1(M^{\circ\circ}), \Delta_B^k \pi_2(M^{\circ\circ}) \rangle$$

and the sequence has no expansion rules.
The last term is equal to $\langle \pi_1(M)^\circ, \pi_2(M)^\circ \rangle = \langle \pi_1(M), \pi_2(M) \rangle^\circ$ and then the property holds.

— If M is of type $A \rightarrow B$ and $M \overset{\eta}{\longrightarrow} \lambda y : A.My$
We know that $\exists k > 0$ such that $M^\circ = \Delta_{A\rightarrow B}^k M^{\circ\circ}$. By corollary 5.7 $\Delta_{A\rightarrow B}^k M^{\circ\circ} \Longrightarrow^+ \lambda y : A.\Delta_B^k(M^{\circ\circ}(\Delta_A^k y))$
and the the sequence has no expansion rules.
The last term is equal to $\lambda y : A.\Delta_B^k(M^{\circ\circ} y^\circ) = \lambda y : A.(My)^\circ = (\lambda y : A.My)^\circ$ and then the property holds.

— If $M : \mathbf{T}$ and $M \overset{Top}{\longrightarrow} *$.
By lemma 5.5 $M^\circ : \mathbf{T}$ and so $M^\circ \overset{Top}{\longrightarrow} * = *^\circ$.

□

Using 5.10 we can show now:

Theorem 5.12. (Simulation) If $\Gamma \vdash M : A$ and $M \Longrightarrow N$, then

1 $\exists k \geq 0$ such that $M^{\circ\circ} \Longrightarrow^{+} \Delta_A^k N^{\circ\circ}$ if not $M \xrightarrow{\eta,\delta} N$
2 $M^{\circ} \Longrightarrow^{+} N^{\circ}$

and there are no expansions in these reduction sequences.

Proof. We show the property by induction on the structure of M. More precisely, for the first statement we analyze each case, while for the second there are two cases:

— if $M \xrightarrow{\eta,\delta} N$, then apply lemma 5.11
— if not $M \xrightarrow{\eta,\delta} N$, then it is enough to analyze only the cases such that $M^{\circ} = M^{\circ\circ}$, because when $M^{\circ} = \Delta_A^h M^{\circ\circ}$ (for $h > 0$) we have easily:

$$M^{\circ} = \Delta_A^h M^{\circ\circ} \Longrightarrow^{+} \text{(by the first statement) } \Delta_A^h \Delta_A^k N^{\circ\circ} = \Delta_A^{h+k} N^{\circ\circ}$$

Then either N is not a pair nor a λ-abstraction, which gives $\Delta_A^{h+k} N^{\circ\circ} = N^{\circ}$ because $h > 0$, or otherwise $\Delta_A^{h+k} N^{\circ\circ} = \Delta_A^{h+k} N^{\circ} \Longrightarrow^{*} N^{\circ}$ by lemma 5.9.

In order to conclude that the reductions in the statements of the lemma contain no expansions, it suffices to notice that every reduction built in the following proof contains no expansion steps: indeed it is constructed from one-step reductions that are not expansions or from reductions obtained by the induction hypothesis (and thus without expansions) or from reductions obtained by lemma 5.11, lemma 5.7, (again without expansions). Now, we can analyze the cases involved in the proof of the first and the second statement.

— $M \equiv *$. It is in normal form.
— $M \equiv x$. The only possible case is $x \xrightarrow{Top} *$, where $x : \mathbf{T}$. Then, $x^{\circ\circ} = x \xrightarrow{Top} * = \Delta_{\mathbf{T}}^0 *^{\circ\circ}$.
— $M \equiv (P_1 Q_1)$.

– If $(P_1 Q_1) : \mathbf{T}$ and $(P_1 Q_1) \xrightarrow{Top} *$, then $(P_1 Q_1)^{\circ\circ} : \mathbf{T}$ by lemma 5.5 and then

$$(P_1 Q_1)^{\circ\circ} \xrightarrow{Top} * = \Delta_{\mathbf{T}}^0 *^{\circ\circ}$$

– If $(\lambda x : C.R) Q_1 \xrightarrow{\beta} R[Q_1/x]$.

$$((\lambda x : C.R) Q_1)^{\circ\circ} = ((\lambda x : C.R)^{\circ\circ} Q_1^{\circ}) = ((\lambda x : C.R^{\circ}) Q_1^{\circ}) \stackrel{\beta}{\Longrightarrow}$$
$$R^{\circ}[Q_1^{\circ}/x] \Longrightarrow^{*} \text{ (by lemma 5.10) } R[Q_1/x]^{\circ} = \Delta_A^k R[Q_1/x]^{\circ\circ}$$

– If $(P_1 Q_1) \Longrightarrow (P_2 Q_1)$, where $P_1 \Longrightarrow P_2$.
Since it is not the case that $P_1 \xrightarrow{\eta} P_2$ because $(P_1 Q_1) \Longrightarrow (P_2 Q_1)$, we have by the induction hypothesis a reduction sequence $P_1^{\circ\circ} \Longrightarrow^{+} \Delta_{B \to A}^h P_2^{\circ\circ}$ without expansions. Then

$$(P_1 Q_1)^{\circ\circ} = P_1^{\circ\circ} Q_1^{\circ} \Longrightarrow^{+} \text{(by ind. hyp.) } (\Delta_{B \to A}^h P_2^{\circ\circ}) Q_1^{\circ}$$

If $h = 0$, then $(P_2^{\circ\circ} Q_1^{\circ}) = (P_2 Q_1)^{\circ\circ}$.

If $h > 0$, then

$$(\Delta^h_{B\to A} P_2^{\circ\circ})Q_1^\circ$$

$$\Downarrow_+ \text{ by lemma 5.7}$$

$$(\lambda w : B.\Delta^h_A(P_2^{\circ\circ}(\Delta^h_B w)))Q_1^\circ$$

$$\downarrow \beta$$

$$\Delta^h_A(P_2^{\circ\circ}(\Delta^h_B Q_1^\circ))$$

$$\Downarrow * \text{ by corollary 5.9}$$

$$\Delta^h_A(P_2^{\circ\circ} Q_1^\circ) = \Delta^h_A(P_2 Q_1)^{\circ\circ}$$

– If $(P_1 Q_1) \Longrightarrow (P_1 Q_2)$, where $Q_1 \Longrightarrow Q_2$

$$(P_1 Q_1)^{\circ\circ} = P_1^{\circ\circ} Q_1^\circ \Longrightarrow^+ \text{(by ind. hyp.)}\ P_1^{\circ\circ} Q_2^\circ = (P_1 Q_2)^{\circ\circ}$$

— $M \equiv \langle P_1, Q_1\rangle$

– If $\langle P_1, Q_1\rangle \Longrightarrow \langle P_2, Q_1\rangle$, where $P_1 \Longrightarrow P_2$.

1 $\langle P_1, Q_1\rangle^{\circ\circ} = \langle P_1^\circ, Q_1^\circ\rangle \Longrightarrow^+$ (by ind. hyp.) $\langle P_2^\circ, Q_1^\circ\rangle = \langle P_2, Q_1\rangle^{\circ\circ} = \Delta^0_A\langle P_2, Q_1\rangle^{\circ\circ}$

2 Since $\langle P, Q\rangle^{\circ\circ} = \langle P, Q\rangle^\circ$, we have $\langle P_1, Q_1\rangle^\circ \Longrightarrow^+ \langle P_2, Q_1\rangle^\circ$ by the previous statement.

– If $\langle P_1, Q_1\rangle \Longrightarrow \langle P_1, Q_2\rangle$, where $Q_1 \Longrightarrow Q_2$.

1 $\langle P_1, Q_1\rangle^{\circ\circ} = \langle P_1^\circ, Q_1^\circ\rangle \Longrightarrow^+$ (by ind. hyp.) $\langle P_1^\circ, Q_2^\circ\rangle = \langle P_1, Q_2\rangle^{\circ\circ} = \Delta^0_A\langle P_1, Q_2\rangle^{\circ\circ}$

2 Since $\langle P, Q\rangle^{\circ\circ} = \langle P, Q\rangle^\circ$, we have $\langle P_1, Q_1\rangle^\circ \Longrightarrow^+ \langle P_1, Q_2\rangle^\circ$ by the previous statement.

— $M \equiv \lambda x : A.P_1$
Then $\lambda x : A.P_1 \Longrightarrow \lambda x : A.P_2$, where $P_1 \Longrightarrow P_2$.

1 $(\lambda x : A.P_1)^{\circ\circ} = \lambda x : A.P_1^\circ \Longrightarrow^+$ (by ind. hyp.) $\lambda x : A.P_2^\circ = \Delta^0_{A\to B}(\lambda x : A.P_2)^{\circ\circ}$

2 Since $(\lambda x : A.P_i)^\circ = (\lambda x : A.P_i)^{\circ\circ}$ we have $(\lambda x : A.P_1)^\circ \Longrightarrow^+ (\lambda x : A.P_2)^\circ$ by the previous statement.

— $M \equiv in^i_C(P_1)$, for $i = 1, 2$ where $P_1 \Longrightarrow P_2$.
Then $in^i_C(P_1) \Longrightarrow in^i_C(P_2)$

$$in^i_C(P_1)^{\circ\circ} = in^i_C(P_1^\circ) \Longrightarrow^+ \text{(by ind. hyp.)}\ in^i_C(P_2^\circ) = \Delta^0_{B+C} in^i_C(P_2)^{\circ\circ}$$

— $M \equiv \pi_i(P_1)$, for $i = 1, 2$.

– If $\pi_i(P_1) : \mathbf{T}$ and $\pi_i(P_1) \xrightarrow{Top} *$, then $\pi_i(P_1)^{\circ\circ} : \mathbf{T}$ by lemma 5.5 and $\pi_i(P_1)^{\circ\circ} \xrightarrow{Top} * = \Delta^0_{Top} *^{\circ\circ}$.

– If $\pi_i(P_1) \Longrightarrow \pi_i(P_2)$, where $P_1 \Longrightarrow P_2$.
Since it is not the case that $P_1 \xrightarrow{\delta} P_2$ because $\pi_i(P_1) \Longrightarrow \pi_i(P_2)$, we have by the induction hypothesis a reduction sequence $P_1^{\circ\circ} \Longrightarrow^+ \Delta^h_{B\times A} P_2^{\circ\circ}$ without expansions. Then

$$\pi_i(P_1)^{\circ\circ} = \pi_i(P_1^{\circ\circ}) \Longrightarrow^+ \text{(by ind. hyp.)}\ \pi_i(\Delta^h_{A_1\times A_2} P_2^{\circ\circ})$$

If $h = 0$, then $\pi_i(P_2^{\circ\circ}) = \pi_i(P_2)^\circ$.
If $h > 0$, then

$$\pi_i(\Delta^h_{A_1\times A_2} P_2^{\circ\circ})$$

$$\Downarrow_+ \text{ by lemma 5.7}$$

$$\pi_i(\langle \Delta^h_{A_1}\pi_1(P_2^{\circ\circ}), \Delta^h_{A_2}\pi_2(P_2^{\circ\circ})\rangle)$$

$$\Downarrow_{\pi_i}$$

$$\Delta^h_{A_i}\pi_i(P_2^{\circ\circ}) = \Delta^h_{A_i}\pi_i(P_2)^{\circ\circ}$$

— $M \equiv Case(P_1, R_1, N_1)$

- If $Case(P_1, R_1, N_1) : \mathbf{T}$ and $Case(P_1, R_1, N_1) \xrightarrow{Top} *$, then $Case(P_1, R_1, N_1)^{\circ\circ} : \mathbf{T}$ by lemma 5.5 and $\pi_i(P_1)^{\circ\circ} \xrightarrow{Top} * = \Delta^0_{Top} *^{\circ\circ}$.

- $Case(in^i_{C_1+C_2}(S), R_1, R_2) \xrightarrow{\rho} R_i S$

$$\begin{array}{l}
Case(in^i_{C_1+C_2}(S), R_1, R_2)^{\circ\circ} = \\
Case(in^i_{C_1+C_2}(S)^\circ, R_1^\circ, R_2^\circ) = \\
Case(in^i_{C_1+C_2}(S^\circ), R_1^\circ, R_2^\circ) \stackrel{\rho}{\Longrightarrow} \\
R_1^\circ S^\circ = \\
(\Delta^h_{C_i\to A} R_1^{\circ\circ}) S^\circ \Longrightarrow^+ \text{(by lemma 5.7)} \\
(\lambda w : C_i.\Delta^h_A(R_1^{\circ\circ}(\Delta^h_{C_i} w)))S^\circ \xrightarrow{\beta} \\
\Delta^h_A(R_1^{\circ\circ}(\Delta^h_{C_i} S^\circ)) \Longrightarrow^* \text{ by corollary 5.9} \\
\Delta^h_A(R_1^{\circ\circ} S^\circ) = \\
\Delta^h_A(R_1 S)^{\circ\circ}
\end{array}$$

- $Case(P_1, R_1, N_1) \Longrightarrow Case(P_2, R_1, N_1)$, where $P_1 \Longrightarrow P_2$.

$$\begin{array}{l}
Case(P_1, R_1, N_1)^{\circ\circ} = \\
Case(P_1^\circ, R_1^\circ, N_1^\circ) \Longrightarrow^+ \text{(by ind. hyp.)} \\
Case(P_2^\circ, R_1^\circ, N_1^\circ) = \\
Case(P_2, R_1, N_1)^{\circ\circ} = \\
\Delta^0_A Case(P_2, R_1, N_1)^{\circ\circ}
\end{array}$$

- $Case(P_1, R_1, N_1) \Longrightarrow Case(P_1, R_2, N_1)$, where $R_1 \Longrightarrow R_2$

$$\begin{array}{l}
Case(P_1, R_1, N_1)^{\circ\circ} = \\
Case(P_1^\circ, R_1^\circ, N_1^\circ) \Longrightarrow^+ \text{(by ind. hyp.)} \\
Case(P_1^\circ, R_2^\circ, N_1^\circ) = \\
Case(P_1, R_2, N_1)^{\circ\circ} = \\
\Delta^0_A Case(P_1, R_2, N_1)^{\circ\circ}
\end{array}$$

- $Case(P_1, R_1, N_1) \Longrightarrow Case(P_1, R_1, N_2)$, where $N_1 \Longrightarrow N_2$

$$
\begin{aligned}
&Case(P_1, R_1, N_1)^{\circ\circ} = \\
&Case(P_1^{\circ}, R_1^{\circ}, N_1^{\circ}) \Longrightarrow^{+} \text{(by ind. hyp.)} \\
&Case(P_1^{\circ}, R_1^{\circ}, N_2^{\circ}) = \\
&Case(P_1, R_1, N_2)^{\circ\circ} = \\
&\Delta_A^0 Case(P_1, R_1, N_2)^{\circ\circ}
\end{aligned}
$$

— $M \equiv (rec\ y : B.P_1)^i$

- If $(rec\ y : \mathbf{T}.P_1)^i : \mathbf{T}$ and $(rec\ y : \mathbf{T}.P_1)^i \xrightarrow{Top} *$, then $((rec\ y : \mathbf{T}.P_1)^i)^{\circ\circ} : \mathbf{T}$ by lemma 5.5 and $((rec\ y : \mathbf{T}.P_1)^i)^{\circ\circ} \xrightarrow{Top} * = \Delta_{Top}^0 *^{\circ\circ}$.
- $(rec\ y : A.P_1)^i \xrightarrow{rec} P_1[(rec\ y : A.P_1)^{i-1}/y]$

$$
\begin{aligned}
&((rec\ y : A.P_1)^i)^{\circ\circ} = \\
&(rec\ y : A.P_1^{\circ})^i \xrightarrow{rec} \\
&P_1^{\circ}[(rec\ y : A.P_1^{\circ})^{i-1}/y] = \\
&P_1^{\circ}[(rec\ y : A.P_1)^{i-1^{\circ\circ}}/y] \Longrightarrow^{*} \text{ (by lemma 5.8)} \\
&(P_1[(rec\ y : A.P_1)^{i-1}/y])^{\circ} = \\
&\Delta_A^h (P_1[(rec\ y : A.P_1)^{i-1}/y])^{\circ\circ}
\end{aligned}
$$

- $(rec\ y : A.P_1)^i \Longrightarrow (rec\ y : A.P_2)^i$.

$$
\begin{aligned}
&((rec\ y : A.P_1)^i)^{\circ\circ} = \\
&(rec\ y : A.P_1^{\circ})^i \Longrightarrow^{+} \text{(by ind. hyp.)} \\
&(rec\ y : A.P_2^{\circ})^i = \\
&\Delta_A^0 ((rec\ y : A.P_2)^i)^{\circ\circ}
\end{aligned}
$$

□

5.2. *Strong Normalization of the Full Calculus*

Having shown that our translation satisfies the hypothesis of Proposition 5.1, all we are now left to prove is that the bounded reduction system without expansion rules is strongly normalizing. This can be established by one of the standard techniques of reducibility, and does not present essential difficulties once the right definitions of *stability* or *reducibility* are given.

In 6 we provide a full proof, adapting Girard's proof from (GLT90), but one can also adapt the proof provided by Poigné and Voss in (PV87), for which we just provide the basic definitions in 6.4. It is then finally possible to state the following

Theorem 5.13. (Strong normalization)
The reduction $\Longrightarrow$ for the bounded system with expansions is strongly normalizing.

Proof. By proposition 5.1, theorem 5.12 and Corollary 6.7. □

6. Strong Normalization without expansion rules

In this section we will prove the strong normalization property for our calculus $\lambda_{\pi*\mu\sigma}$, with labeled recursion, but no expansions, using the *reducibility* method as in (GLT90), with an additional astute twist to take care of the sum type and labeled recursion.

6.1. *Reducibility*

We define the set RED_A of *reducible* terms of type A by induction on the type A as follows:

— For M of atomic type A, $M \in RED_A$ iff M is strongly normalizable
— For M of product type, $M \in RED_{A_1 \times A_2}$ iff $\pi_i(M) \in RED_{A_i}$
— For M of a sum type, $M \in RED_{A_1+A_2}$ iff, for fresh variables $w_i : A_i$, we have $Case(M, \lambda x : A_1.\langle x, w_2\rangle, \lambda y : A_2.\langle w_1, y\rangle) \in RED_{A_1 \times A_2}$ (in the case A_i is **T**, we take $*$ instead of w_i)
— For M of a functional type, $M \in RED_{A_1 \to A_2}$ iff for all $N \in RED_{A_1}$, $(MN) \in RED_{A_2}$

Some comment on the sum type are needed here: first of all notice that the notion of reducibility is well defined: reducibility for a sum type is given in term of reducibility for a product type, which has been defined before. Secondly, notice that for all other types, reducibility is either given directly as in the case of the base types, or given in terms of reducibility for types that are strictly smaller. This is not possible for the sum type, because we have no destructor associated to it, but only a case expression, so reducibility for $A + B$ really depends on reducibility of A and B *together*, and we express this fact by reducing it to reducibility of the product $A \times B$.

6.2. *Properties of reducibility*

Following (GLT90), we define a notion of *neutrality*: a term is *neutral* if does not interact with the surrounding context giving raise to redexes. In our case, the neutral terms are:

$$* \quad x \quad \pi_i(M) \quad Case(P, M, N) \quad (MN) \quad (rec\, y.M)^i$$

We will prove that RED_A enjoys the following properties, for all types A:

(CR1) If $M \in RED_A$, then M is strongly normalizable.
(CR2) If $M \in RED_A$, and M reduces to M', then $M' \in RED_A$.
(CR3) If M is neutral and whenever we perform on it one step of reduction we obtain a term $M' \in RED_A$, then $M \in RED_A$.
As a special case of the last clause:

(CR4) If M is neutral and no reduction is applicable to it, then $M \in RED_A$.

In particular, $*$ and the variables are reducible (also the variables of type **T**, as they can only reduce to $*$, which is reducible).

Proposition 6.1. (Properties of reducibility) For every type A, the set RED_A satisfies **(CR1)**, **(CR2)** and **(CR3)**.

Proof. We will proceed by induction on the type A.

6.2.1. *Atomic types*

(CR1) A reducible term of atomic type is strongly normalizable by definition
(CR2) If M is strongly normalizable, then so is every reduct of M (as reduction preserves the type)
(CR3) Suppose all one step reducts of M are reducible, i.e. strongly normalizable. Any reduction path leaving M must pass through one of its one-step reducts, which are in a finite number, so that the longest reduction sequence starting from M has length the maximum among the $1 + \nu(M')$, as M' varies over the (one-step) reducts of M. Since these lengths are all finite, M is strongly normalizing.

6.2.2. *Product types*

(CR1) Suppose $M \in RED_{A_1 \times A_2}$. Then by definition we know that $\pi_i(M)$ are reducible and so strongly normalizing by the induction hypothesis. This implies that M is strongly normalizing also, because any reduction sequence starting from M can be turned into a reduction sequence starting from $\pi_i(M)$.

(CR2) We know that $\pi_i(M) \in RED_{A_i}$, by definition. Now consider the possible one step reducts of M:

— M reduces to M'. Then also $\pi_i(M)$ reduces to $\pi_i(M')$ via the same reduction, and M' is then reducible by definition

(CR3) Let now M be neutral (not necessarily reducible) such that all its one step reducts are reducible. We must show that $\pi_i(M)$ is reducible of type A_i. Since M cannot be a pair (as it is neutral), any one step reduction of $\pi_i(M)$ must be to a term $\pi_i(M')$, with M' one step from M. By **(CR2)**, M' is reducible, and then by definition also $\pi_i(M')$ is reducible. Now, $\pi_i(M)$ is neutral and all its one step reducts are reducible, hence by the induction hypothesis **(CR3)** for A_i, $\pi_i(M)$ is reducible, hence M is, by definition.

6.2.3. *Arrow types*

(CR1) Suppose $M \in RED_{A_1 \to A_2}$. Then by definition we know that MN is reducible for all reducible N. In particular, Mx is reducible for a fresh variable x, which is reducible by the induction hypothesis **(CR3)** for A_1, hence Mx is strongly normalizable. This implies that also M is strongly normalizing, as all reduction sequences starting from M can be performed also on Mx.

(CR2) Let $M \in RED_{A_1 \to A_2}$ reduce to M'. For all $N \in RED_{A_1}$ we have $(M'N) \in RED_{A_2}$, since it is a reduct of (MN), which is reducible because M and N are. Hence M' is reducible by definition.

(CR3) Let now M be neutral (not necessarily reducible) such that all one step reductions lead to reducible terms. We show that MN is reducible for all reducible N by induction on $\nu(N)$, using **(CR3)** for A_2. Consider a one step reduction of MN: since M is neutral, this reduction must be either inside M or inside N and leads to:

— $M'N$, with M' one step from M, so M' is reducible and hence $M'N$ is

— MN', with N' one step from N; N' is reducible by **(CR2)** for A_1, and $\nu(N') < \nu(N)$, so by the induction hypothesis MN' is reducible

Hence all reductions leaving MN lead to a reducible term and hence MN is reducible for all reducible N, so that M is reducible by definition.

6.2.4. *Sum types*

(CR1) Suppose $M \in RED_{A_1+A_2}$. Then by definition $Case(M, \lambda x.\langle x, w_2\rangle, \lambda y.\langle w_1, y\rangle) \in RED_{A_1 \times A_2}$ is reducible, hence strongly normalizable, hence M is strongly normalizable too.

(CR2) Suppose $M \in RED_{A_1+A_2}$ reduces to M'. Then $Case(M, \lambda x.\langle x, w_2\rangle, \lambda y.\langle w_1, y\rangle) \in RED_{A_1 \times A_2}$ reduces to $Case(M', \lambda x.\langle x, w_2\rangle, \lambda y.\langle w_1, y\rangle)$, so that, by **(CR2)** for $A \times B$ which has been proved before, $Case(M, \lambda x.\langle x, w_2\rangle, \lambda y.\langle w_1, y\rangle) \in RED_{A_1 \times A_2}$. Hence M' is reducible too.

(CR3) Let now M be neutral, and suppose all its one step reducts are reducible. We will show that $Case(M, \lambda x.\langle x, w_2\rangle, \lambda y.\langle w_1, y\rangle)$ (which is neutral) is reducible using **(CR3)** for $A_1 \times A_2$, which has already been proved to hold. Consider the possible one step reducts:

— $Case(M', \lambda x.\langle x, w_2\rangle, \lambda y.\langle w_1, y\rangle)$ with M' one step from M: then M' is reducible, hence by definition $Case(M', \lambda x.\langle x, w_2\rangle, \lambda y.\langle w_1, y\rangle)$ is reducible

— there is no other one step reduct as M is neutral and the terms $\lambda x.\langle x, w_2\rangle$ and $\lambda y.\langle w_1, y\rangle$ are normal

□

6.3. *Reducibility theorem*

We are left to show a few more lemmas:

Lemma 6.2. (Pairing) Let $M_1 : A_1$, $M_2 : A_2$ be reducible terms. Then $\langle M_1, M_2\rangle \in RED_{A_1\times A_2}$.

Proof. We need to show that $\pi_i(\langle M_1, M_2\rangle) \in RED_{A_i}$.
Since $\pi_i(\langle M_1, M_2\rangle)$ is neutral, we prove the statement using **(CR3)**: we will show that all one step reductions are reducible. We proceed by induction on the sum $\nu(M_1) + \nu(M_2)$ of the maximum reduction lengths for M_1 and M_2, (which are finite, as these terms are strongly normalizable by **(CR1)**).
The possible reducts are:

— M_i, which is reducible by hypothesis
— $\pi_i(\langle M_1', M_2\rangle)$: now, M_1' is one step from M_1, so that $\nu(M_1') + \nu(M_2) < \nu(M_1) + \nu(M_2)$, and M_1' is reducible by **(CR2)**, so $\pi_i(\langle M_1', M_2\rangle)$ is reducible by the induction hypothesis
— $\pi_i(\langle M_1, M_2'\rangle)$: this is shown reducible as the term in the previous case

□

Lemma 6.3. (Abstraction) Let $M : A_2$ be a term where the variable $x : A_1$ may occur free. If for every $N \in RED_{A_1}$ we have $M[N/x] \in RED_{A_2}$, then $\lambda x : A_1.M \in RED_{A_1\to A_2}$.

Proof. We want to show that $(\lambda x.M)P$ is reducible for all reducible P. Since this term is neutral, we can prove our Lemma using **(CR3)**. We are then left to show that all one step reducts of $(\lambda x.M)P$ are reducible if for all $N \in RED_{A_1}$ we have $M[N/x] \in RED_{A_2}$. Since $M = M[x/x]$ is reducible by hypothesis (as any variable is reducible), it is strongly normalizable by **(CR1)**, and we can proceed to prove this last statement by induction on $\nu(M) + \nu(P)$. The term $(\lambda x.M)P$ converts to:

— $M[P/x]$ which is reducible by hypothesis
— $(\lambda x.M')P$ with M' a reduct of M; now, by **(CR2)**, M' is still reducible and furthermore $\nu(M') + \nu(P) < \nu(M)+\nu(P)$ and $M[P/x]$ reduces to $M'[P/x]$, and this last term is also reducible, because it is a multi-step reduct of $M[P/x]$ by Lemma 4.5. So the induction hypothesis tells us that $(\lambda x.M')P$ is reducible.
— $(\lambda x.M)P'$ with P' a reduct of P; now, by **(CR2)**, P' is still reducible and furthermore $\nu(M) + \nu(P') < \nu(M) + \nu(P)$ and $M[P/x]$ reduces to $M[P'/x]$, by Corollary 4.8, so this last term is also reducible. The induction hypothesis tells us that $(\lambda x.M)P'$ is reducible.

□

Lemma 6.4. (Injections) For all terms, $M \in RED_{A_1}$ iff $in^i_{A_1+A_2}(M) \in RED_{A_1+A_2}$.

Proof. ($\Rightarrow$)
We must show that $Case(in^i_{A_1+A_2}(M), \lambda x.\langle x, w_2\rangle, \lambda y.\langle w_1, y\rangle)$ is reducible of type $A_1\times A_2$, i.e. that $\pi_i(Case(in^i_{A_1+A_2}(M), \lambda x.\langle x, w_2\rangle, \lambda y.\langle w_1, y\rangle)) \in RED_{A_i}$. We will proceed using **(CR3)**, by induction on $\nu(M)$, because $\pi_i(Case(in^i_{A_1+A_2}(M), \lambda x.\langle x, w_2\rangle, \lambda y.\langle w_1, y\rangle))$
is neutral.
Consider then all its one step reducts:

— $((\lambda x.\langle x, w_2\rangle)M)$, which is reducible because M is reducible and $\lambda x.\langle x, w_2\rangle$ is reducible (by Lemma 6.2 applied to the variables x and w_2 we know that $\langle x, w_2\rangle$ is reducible, and we get reducibility of $\lambda x.\langle x, w_2\rangle$ by Lemma 6.3; similarly if we have $*$ instead of w_2)
— $\pi_i(Case(in^i_{A_1+A_2}(M'), \lambda x.\langle x, w_2\rangle, \lambda y.\langle w_1, y\rangle))$ with M' one step from M, hence reducible and $\nu(M') < \nu(M)$, so that it is reducible by the induction hypothesis

($\Leftarrow$)
Suppose now, $in^i_{A_1+A_2}(M) \in RED_{A_1+A_2}$. This means, by definition of reducibility over sum types,

$$Case(in^i_{A_1+A_2}(M), \lambda x.\langle x, w_2\rangle, \lambda y.\langle w_1, y\rangle) \in RED_{A_1\times A_2},$$

which implies, by definition of reducibility over product types,

$$\pi_1(Case(in^i_{A_1+A_2}(M), \lambda x.\langle x, w_2\rangle, \lambda y.\langle w_1, y\rangle)) \in RED_{A_1}$$

This term reduces to $\pi_1((\lambda x.\langle x, w_2\rangle M))$, which in turn reduces to $\pi_1(\langle M, w\rangle)$ and then to M, which is then reducible by repeated use of **(CR2)**. □

Lemma 6.5. (Sum) Let $P : A + B$, $M : A \to C$ and $N : B \to C$ be reducible terms. Then $Case(P, M, N) \in RED_C$.

Proof. We will work by cases on C.

C is an atomic type We can use **(CR3)** for C, as $Case(P, M, N)$ is neutral. We will show by induction on $\nu(P) + \nu(M) + \nu(N)$ that all one step reducts of $Case(P, M, N)$ are reducible. Consider the possible one step reducts:

— $Case(P', M, N)$, or $Case(P, M', N)$, or $Case(P, M, N')$: they are reducible by the induction hypothesis as all primed terms are reducible by **(CR2)** on $A + B$, $A \to C$, $B \to C$, and the measure decreases strictly.

— (RM) if $P \equiv in^i_{A_1+A_2}(R)$: then R is also reducible by Lemma 6.4, and this term is reducible as M is

$C \equiv C_1 \times C_2$ We must show $\pi_i(Case(P, M, N)) \in RED_{C_i}$. Since $\pi_i(Case(P, M, N))$ is neutral, we can use **(CR3)** for C_i. Since P, M, N, are all reducible, they are all strongly normalizable and we can proceed by induction on the measure $\nu(P) + \nu(M) + \nu(N)$. Consider the possible one step reducts:

— $\pi_i(Case(P', M, N))$ or $\pi_i(Case(P, M', N))$ or $\pi_i(Case(P, M, N'))$: they are reducible by the induction hypothesis as all primed terms are reducible by **(CR2)** on $A + B$, $A \to C$, $B \to C$, and the measure decreases strictly.

— $\pi_i((MR))$ if $P \equiv in^i_{A_1+A_2}(R)$: then R is also reducible by Lemma 6.4, so MR is reducible and $\pi_i((MR))$ too

$C \equiv C_1 \to C_2$ We must show $Case(P, M, N)Q \in RED_{C_2}$ for all $Q \in RED_{C_1}$. Since $Case(P, M, N)Q$ is neutral, we can use **(CR3)** for C_2. Since P, M, N, Q are all reducible, they are all strongly normalizable and we can proceed by induction on the measure $\nu(P) + \nu(M) + \nu(N) + \nu(Q)$. Consider the possible one step reducts:

— $Case(P', M, N)Q$, or $Case(P, M', N)Q$, or $Case(P, M, N')Q$, or $Case(P, M, N)Q'$: they are reducible by the induction hypothesis as all primed terms are reducible by **(CR2)** on $A + B$, $A \to C$, $B \to C$ and C_1, and the measure decreases strictly.

— $(RM)Q$ if $P \equiv in^i_{A_1+A_2}(R)$: then R is also reducible by Lemma 6.4, and this term is reducible as M and Q are

$C \equiv C_1 + C_2$ We must show $Case(Case(P, M, N), \lambda x.\langle x, w_2\rangle, \lambda y.\langle w_1, y\rangle) \in RED_{C_1\times C_2}$. We can use **(CR3)** for $C_1 \times C_2$ because $Case(Case(P, M, N), \lambda x.\langle x, w_2\rangle, \lambda y.\langle w_1, y\rangle)$ is neutral. Since P, M, N, are all reducible, they are all strongly normalizable and we can proceed by induction on the measure $\nu(P) + \nu(M) + \nu(N)$. Consider the possible one step reducts:

— $Case(Case(P', M, N), \lambda x.\langle x, w_2\rangle, \lambda y.\langle w_1, y\rangle)$ or $Case(Case(P, M', N), \lambda x.\langle x, w_2\rangle, \lambda y.\langle w_1, y\rangle)$ or $Case(Case(P, M, N'), \lambda x.\langle x, w_2\rangle, \lambda y.\langle w_1, y\rangle)$: they are reducible by the induction hypothesis as all primed terms are reducible by **(CR2)** on $A + B$, $A \to C$, $B \to C$, and the measure decreases strictly.

— $Case((RM), \lambda x.\langle x, w_2\rangle, \lambda y.\langle w_1, y\rangle)$ if $P \equiv in^i_{A_1+A_2}(R)$: then R is also reducible by Lemma 6.4, and this term is reducible by definition as M, hence also (RM), is

□

We will now prove that every reducible instance of a (not necessarily reducible) term M is reducible. As a consequence, all terms will be reducible.

Theorem 6.6. (Reducibility) Let M be *any* term (not assumed to be reducible), and suppose all the free variables of M are among $x_1, \ldots x_n$ of types $A_1, \ldots A_n$. If $N_1, \ldots N_n$ are reducible terms of types $A_1, \ldots A_n$, then $M[\overline{N}/\overline{x}]$ is reducible.

Proof. By induction on the structure of M.

1 M is $*$. It is neutral and normal, so it is reducible.
2 M is x_i for some i, then $M[\overline{N}/\overline{x}] = N_i$ is reducible
3 $M \equiv \pi_i(M')$. By the induction hypothesis, $M'[\overline{N}/\overline{x}]$ is reducible, hence, by definition, $\pi_i(M'[\overline{N}/\overline{x}]) = \pi_i(M')[\overline{N}/\overline{x}]$ is reducible.
4 $M \equiv \langle M_1, M_2\rangle$. By the induction hypothesis, the terms $M_i[\overline{N}/\overline{x}]$ are reducible, so we conclude that the term $\langle M_1[\overline{N}/\overline{x}], M_2[\overline{N}/\overline{x}]\rangle = \langle M_1, M_2\rangle[\overline{N}/\overline{x}]$ is reducible.
5 $M \equiv in^i_{A_1+A_2}(M')$. By the induction hypothesis, $M'[\overline{N}/\overline{x}]$ is reducible, hence by Lemma 6.4, we have that $in^i_{A_1+A_2}(M'[\overline{N}/\overline{x}]) = in^i_{A_1+A_2}(M')[\overline{N}/\overline{x}]$ is reducible.
6 $M \equiv Case(M_1, M_2, M_3)$. By the induction hypothesis, the terms $M_i[\overline{N}/\overline{x}]$ are reducible, hence we have that $Case(M_1[\overline{N}/\overline{x}], M_2[\overline{N}/\overline{x}], M_3[\overline{N}/\overline{x}]) = Case(M_1, M_2, M_3)[\overline{N}/\overline{x}]$ is reducible.
7 $M \equiv (M_1 M_2)$. By the induction hypothesis, the terms $M_i[\overline{N}/\overline{x}]$ are reducible, so we conclude that the term $(M_1[\overline{N}/\overline{x}] M_2[\overline{N}/\overline{x}]) = (M_1 M_2)[\overline{N}/\overline{x}]$ is reducible.
8 $M \equiv \lambda y.M'$. Then by the induction hypothesis $M'[\overline{N}/\overline{x}][N'/y]$ is reducible for all reducible terms N'. By Lemma 6.3, $\lambda y.M'[\overline{N}/\overline{x}] = (\lambda y.M')[\overline{N}/\overline{x}]$ is reducible.
9 $M \equiv (recy.M')^i$. By the induction hypothesis, $M'[\overline{N}/\overline{x}]$ is reducible. We will show reducibility for $(recy.M')^i[\overline{N}/\overline{x}]$ by induction on $i + \nu(M')$. Since $(recy.M)^i[\overline{N}/\overline{x}]$ is neutral, we will use **(CR3)** for the type A of $(recy.M)^i$. Consider the one step reducts of $(recy.M)^i[\overline{N}/\overline{x}]$

— $(recy.M'')^i[\overline{N}/\overline{x}]$ with M'' one step from M'. Then $M''[\overline{N}/\overline{x}]$ is reducible for all reducible $\overrightarrow{N}$, because it is a multi-step reduct of the reducible term $M'[\overline{N}/\overline{x}]$ (Lemma 4.5). Furthermore,$i+\nu(M'') < i + \nu(M')$, so by the induction hypothesis $(recy.M''[\overline{N}/\overline{x}])^i = (recy.M'')^i[\overline{N}/\overline{x}]$ is reducible.

— $M'[\overline{N}/\overline{x}][(recy.M'[\overline{N}/\overline{x}])^{i-1}/y]$. Then $(recy.M')^{i-1}[\overline{N}/\overline{x}] = (recy.M'[\overline{N}/\overline{x}])^{i-1}$ is reducible by the induction hypothesis, and this tells us that $[\overline{N}/\overline{x}][(recy.M'[\overline{N}/\overline{x}])^{i-1}/y]$ is a substitution of reducible terms for a set of variables containing the free variables of M', which gives us reducibility of the term $M'[\overline{N}/\overline{x}][(recy.M'[\overline{N}/\overline{x}])^{i-1}/y]$.

□

Corollary 6.7. (Strong Normalization) All terms are reducible, hence strongly normalizable.

6.4. *Another method*

It is also possible to adapt a proof based on the notion of *stability*, as the one provided by Poingé and Voss. We just give here the basic definition, and refer the interested reader to (DCK93b) for full details.

6.5. *Stability*

We define a set of *stable terms of type A* by induction on the type A in the following way:

— For M of atomic type A, M is stable if and only if it is strongly normalizing.

— For M of type $A_1 \times A_2$, M is stable if and only if it is strongly normalizing and whenever M reduces to $\langle M_1, M_2 \rangle$, then M_1 and M_2 are stable terms of type A_1 and A_2 respectively.

— For M of type $A_1 + A_2$, M is stable if and only if it is strongly normalizing and whenever M reduces to $in^i_{A_1+A_2}(M')$ then M' is stable of type A_i.

— For M of type $A_1 \to A_2$, M is stable if and only if for every stable term N of type A_1, MN is a stable term of type A_2.

7. Confluence of the Full Calculus

In this section we deduce the confluence property for the calculus with bounded recursion as well as for the version with unbounded recursion.

We can immediately deduce the confluence property for the bounded system from the weak confluence and strong normalization properties using Newman's Lemma, however, we can also provide an extremely simple and neat proof that does not need the weak confluence property for the expansionary system.

Theorem 7.1. (Confluence) The relation $\Longrightarrow$ is Church-Rosser.

Proof. Since $\Longrightarrow$ is weakly confluent by theorem 4.11 and strongly normalizing by theorem 5.13 we can conclude that it is Church Rosser by the well known Newman's lemma.
The other proof of confluence proceeds as follows.
Let M be a term s.t. $P_1 \;{}^*\!\!\Longleftarrow M \Longrightarrow^* P_2$. Since $\Longrightarrow$ is strongly normalizing, we can reduce the terms P_i to their normal forms $\overline{P_i}$. Then we have $\overline{P_1} \;{}^*\!\!\Longleftarrow M \Longrightarrow^* \overline{P_2}$, and by theorem 5.12 $\overline{P_1}^\circ \;{}^+\!\!\Longleftarrow M^\circ \Longrightarrow^+ \overline{P_2}^\circ$ without expansions in the reduction sequences. As the system without expansions is confluent (we showed that it is strongly normalizing, and weak confluence without expansions can be shown as easily as for the simply typed lambda calculus), we can close the internal diagram with $\overline{P_1}^\circ \Longrightarrow^* R \;{}^*\!\!\Longleftarrow \overline{P_2}^\circ$. Now, $\overline{P_i}^\circ =_{lemma\ 5.6} \overline{P_i}$ and therefore we can complete the proof using the reductions $P_1 \Longrightarrow^* \overline{P_1} \Longrightarrow^* R \;{}^*\!\!\Longleftarrow \overline{P_2} \;{}^*\!\!\Longleftarrow P_2$ (notice that $\overline{P_1} = R = \overline{P_2}$). The following figure shows the reduction diagram:

$$
\begin{tikzcd}
 & M \arrow[dl, "*"'] \arrow[dr, "*"] & \\
P_1 \arrow[d, "*"] & & P_2 \arrow[d, "*"] \\
\overline{P_1} & M^\circ \arrow[ddl, "*"'] \arrow[ddr, "*"] & \overline{P_2} \\
\equiv & & \equiv \\
\overline{P_2}^\circ \arrow[dr, "*"'] & & \overline{P_2}^\circ \arrow[dl, "*"] \\
 & R &
\end{tikzcd}
$$

□

In order to show confluence of the full calculus we relate in the first place the bounded reduction $\Longrightarrow$ and the unbounded one $\stackrel{\infty}{\Longrightarrow}$, and then we use the confluence of $\Longrightarrow$ to show the confluence of $\stackrel{\infty}{\Longrightarrow}$. This very same technique, that originates from early work of Lévy (Lév76), was used in (PV87). The connection between the reductions $\Longrightarrow$ and $\stackrel{\infty}{\Longrightarrow}$ comes from the following:

Remark 7.2. If $M \Longrightarrow^* N$, then $|M| \stackrel{\infty}{\Longrightarrow} |N|$, where $|M|$ is obtained from M by removing all the indices from the *rec* terms.

Lemma 7.3. For any reduction sequence $M_0 \stackrel{\infty}{\Longrightarrow} M_1 \stackrel{\infty}{\Longrightarrow} \ldots \stackrel{\infty}{\Longrightarrow} M_n$, there exists an indexed computation $N_0 \Longrightarrow N_1 \Longrightarrow \ldots \Longrightarrow N_n$ such that $|N_i| = M_i$, for $i = 0 \ldots n$.

Proof. Index all the rec constructors in M_0 by a number $n + k$, with $k \geq 0$. □

Confluence of the full calculus results now from the confluence of the bounded calculus.

Theorem 7.4. $\stackrel{\infty}{\Longrightarrow}$ is Church Rosser.

Proof. Let $M \equiv P_0 \stackrel{\infty}{\Longrightarrow} P_1 \stackrel{\infty}{\Longrightarrow} \ldots \stackrel{\infty}{\Longrightarrow} P_n$ and $M \equiv Q_0 \stackrel{\infty}{\Longrightarrow} Q_1 \stackrel{\infty}{\Longrightarrow} \ldots \stackrel{\infty}{\Longrightarrow} Q_m$. By lemma 7.3 $P_0^{'} \Longrightarrow P_1^{'} \Longrightarrow \ldots \Longrightarrow P_n^{'}$ and $Q_0^{'} \Longrightarrow Q_1^{'} \Longrightarrow \ldots \Longrightarrow Q_m^{'}$ are indexed computations, where $|P_i^{'}| = P_i$, for $i = 0 \ldots n$ and $|Q_i^{'}| = Q_i$, for $i = 0 \ldots m$.
We can assume that $P_0^{'} \equiv Q_0^{'}$ by indexing P^0 and Q^0 with $n+m$. As $\Longrightarrow$ is Church Rosser by theorem 7.1, there exists a term R such that $P_n^{'} \Longrightarrow^* R$ and $Q_m^{'} \Longrightarrow^* R$. By 7.2 $P_n = |P_n^{'}| \stackrel{\infty}{\Longrightarrow}^* |R|$ and $Q_m = |Q_m^{'}| \stackrel{\infty}{\Longrightarrow}^* |R|$. □

8. Adding weak extensionality for the sum type

In this section we show how to apply our techniques in order to accommodate in our system the weak extensionality for the sum type, that is described by the following equality, which tells us that any term P of sum type $A_1 + A_2$ is definitely an injection from one of the two types A_i.

$$Case(P, \lambda x.in^1(x), \lambda y.in^2(y)) = P \tag{1}$$

This is the usual equality that is found in proof theory, associated to the logical connective for disjunction (see for example (GLT90; Gir72)). We call this rule "weak" because in category theory there is another stronger kind of extensional equality associated to the sum, that is used to axiomatize the uniqueness of the sum of two arrows in the diagram for the coproduct, namely

$$Case(P, M \circ \lambda x.in^1(x), M \circ \lambda y.in^2(y)) = MP \tag{2}$$

where $M \circ N$ is the usual abbreviation for the composition $\lambda x.(M(Nx))$.

One can easily see that this strong rule really breaks down into two simpler rules: the weak rule 1 we just introduced and the following commutation rule:

$$Case(P, M \circ N_1, M \circ N_2) = MCase(P, N_1, N_2) \tag{3}$$

If one really wants the equality 2, it seems to be a difficult task to provide a confluent system for the extensional theory with arrow, product and coproduct types, as discussed in (Dou90), and to the author's best knowledge, there are no positive results in that direction.

Notice also that the equation 1 can be easily added to a reduction system with no **T** type, where all the extensional equalities are turned into *contractions*, as done for example in (Gal93). In the presence of the **T** type, to use contraction rules one is forced to proceed along the lines of (CDC91), and to generate an infinite set of reduction rules.

It is not obvious to add weak extensionality for the sum to our system, as the naïve idea of adding the equality 1 as a contraction rule breaks confluence, as the following example shows:

$$\begin{array}{c} Case(w, \lambda x : A \to B.in^1_{(A\to B)+C}(x), \lambda y : C.in^2_{(A\to B)+C}(y)) \Longrightarrow w \\ \Downarrow \\ Case(w, \lambda x : A \to B.in^1_{(A\to B)+C}(\lambda z : A.xz), \lambda y : C.in^2_{(A\to B)+C}(y)) \end{array}$$

This problem comes from the fact that the term $\lambda x : A \to B.in^1_{(A\to B)+C}(x)$ is not in normal form w.r.t. the rules η, δ and Top. This also suggests the solution: it suffices to completely expand the terms $IN^1 = \lambda x.in^1(x)$ and $IN^2 = \lambda y.in^2(y)$ w.r.t. the rules η, δ and Top (which we know now are strongly normalizing) before performing the contraction for the weak sum extensional equality.

So we are led to consider the contraction rule:

$$Case(P, \|IN^1\|, \|IN^2\|) \xrightarrow{+} P$$

where $\|M\|$ denotes the normal form of M w.r.t. η, δ and Top expansions.

It is now straightforward to check that the weak confluence property still holds, and one is left to check that the simulation theorem stays valid.

For that, we have to verify that $Case(P, \|IN^1\|, \|IN^2\|)^\circ \Longrightarrow^+ P^\circ$ without using expansion rules in the reduction sequence, and this is obtained by:

$$\begin{array}{l} Case(P, \|IN^1\|, \|IN^2\|)^\circ = \\ \Delta^k_{A+B} Case(P^\circ, \|IN^1\|^\circ, \|IN^2\|^\circ) = \\ \Delta^k_{A+B} Case(P^\circ, \|IN^1\|, \|IN^2\|) \stackrel{+}{\Longrightarrow} \\ \Delta^k_{A+B} P^\circ \Longrightarrow^* \\ P^\circ \end{array}$$

Notice that the rules η, δ and Top do not create new redexes, as shown in lemma 3.7, so in particular $\|IN^i\|$ is still in normal form, and the equality $\|IN^i\|^\circ = \|IN^i\|$ can be obtained from lemma 5.6.

9. Conclusions

We have provided a confluent rewriting system for an extensional typed λ-calculus with product, sum, terminal object and recursion, which is also strongly normalizing in case the recursion operator is bounded. There are mainly two relevant technical contributions in this paper: the weak confluence proof and the simulation theorem.

On one hand, let us remark once again that the weak confluence property for a context-sensitive reduction system is not as straightforward as for the reduction systems that are congruencies. The proof is no longer just a matter of a boring but trivial case analysis, so we had to explore and analyze here the fine structure of the reduction system, showing clearly how substitution and reduction interact in the presence of context-sensitive rules.

The simulation theorem, on the other hand, turns out to be the real key tool for this expansionary system: it allows to reduce *both* confluence *and* strong normalization properties to those for the underlying calculus without expansions, that can be proved using the standard techniques. In a sense, this is all that you really need to prove.

References

Y. Akama. On mints' reductions for ccc-calculus. In M. Bezem and J. Groote, editors, *Typed Lambda Calculus and Applications*, volume 664 of *LNCS*, pages 1–12. Springer Verlag, 1993.

H. Barendregt. *The Lambda Calculus; Its syntax and Semantics (revised edition)*. North Holland, 1984.

P.-L. Curien and R. Di Cosmo. A confluent reduction system for the λ-calculus with surjective pairing and terminal object. In Leach, Monien, and Artalejo, editors, *Intern. Conf. on Automata, Languages and Programming (ICALP)*, number 510 in LNCS, pages 291–302. Springer-Verlag, 1991.

H. Curry and R. Feys. *Combinatory Logic*, volume 1. North Holland, 1958.

D. Cubric. On free ccc. Distributed on the `types` mailing list, 1992.

R. Di Cosmo and D. Kesner. A confluent reduction for the extensional typed $\lambda-$calculus with pairs, sums, recursion and terminal object. In A. Lingas, editor, *Intern. Conf. on Automata, Languages and Programming (ICALP)*, LNCS. Springer-Verlag, 1993.

R. Di Cosmo and D. Kesner. Simulating expansions without expansions. Technical Report LIENS-93-11/INRIA 1911, LIENS-DMI and INRIA, 1993.

D. J. Dougherty. Some reduction properties of a lambda calculus with coproducts and recursive types. Technical report, Wesleyan University, 1990. E-mail: `ddougherty@eagle.wesleyan.edu`.

D. J. Dougherty. Some lambda calculi with categorical sums and products. In *Proc. of the Fifth International Conference on Rewriting Techniques and Applications (RTA)*, 1993.

J. Gallier. On the correspondence between proofs and λ-terms. Available by anonymous ftp from ftp.cis.upenn.edu. Author's address: jean@saul.cis.upenn.edu., January 1993.

J.-Y. Girard. *Interprétation fonctionelle et élimination des coupures dans l'arithmétique d'ordre supérieure*. Thèse de doctorat d'état, Université de Paris VII, 1972.

J.-Y. Girard, Y. Lafont, and P. Taylor. *Proofs and Types*. Cambridge University Press, 1990.

G. Huet. Résolution d'équations dans les langages d'ordre $1, 2, \ldots, \omega$. *Thèse d'Etat, Université Paris VII*, 1976.

C. B. Jay. Long $\beta\eta$ normal forms and confluence (and its revised version). Technical Report ECS-LFCS-91-183, LFCS, University of Edimburgh, 1992.

C. B. Jay and N. Ghani. The virtues of eta-expansion. Technical Report ECS-LFCS-92-243, LFCS, University of Edimburgh, 1992.

J. W. Klop. Combinatory reduction systems. *Mathematical Center Tracts*, 27, 1980.

J.-J. Lévy. An algebraic interpretation of the $\lambda\beta\kappa$-calculus and a labelled λ-calculus. *Theoretical Computer Science*, 2:97–114, 1976.

J. Lambek and P. J. Scott. *An introduction to higher order categorical logic*. Cambridge University Press, 1986.

G. Mints. Closed categories and the theory of proofs. *Zapiski Nauchnykh Seminarov Leningradskogo Otdeleniya Matematicheskogo Instituta im. V.A. Steklova AN SSSR*, 68:83–114, 1977.

G. Mints. Teorija categorii i teoria dokazatelstv.I. *Aktualnye problemy logiki i metodologii nauky*, pages 252–278, 1979.

G. Pottinger. The Church Rosser Theorem for the Typed lambda-calculus with Surjective Pairing. *Notre Dame Journal of Formal Logic*, 22(3):264–268, 1981.

D. Prawitz. Ideas and results in proof theory. *Proceedings of the 2nd Scandinavian Logic Symposium*, pages 235–307, 1971.

A. Poigné and J. Voss. On the implementation of abstract data types by programming language constructs. *Journal of Computer and System Science*, 34(2-3):340–376, April/June 1987.

A. S. Troelstra. Strong normalization for typed terms with surjective pairing. *Notre Dame Journal of Formal Logic*, 27(4), 1986.