J. Functional Programming 6 (2): 299–327, March 1996 © 1996 Cambridge University Press

A confluent reduction for the λ-calculus with surjective pairing and terminal object

PIERRE-LOUIS CURIEN
LIENS (CNRS URA 1327) - DMI

ROBERTO DI COSMO
LIENS (CNRS URA 1327) - DMI,
and Dipartimento di Scienze dell'Informazione, Pisa, Italy

Abstract

We exhibit confluent and effectively weakly normalizing (thus decidable) rewriting systems for the full equational theory underlying cartesian closed categories, and for polymorphic extensions of it. The λ-calculus extended with surjective pairing has been well-studied in the last two decades. It is not confluent in the untyped case, and confluent in the typed case. But to the best of our knowledge the present work is the first treatment of the lambda calculus extended with surjective pairing *and* terminal object via a *confluent* rewriting system, and is the first solution to the decidability problem of the full equational theory of Cartesian Closed Categories extended with *polymorphic types*. Our approach yields conservativity results as well. In separate papers we apply our results to the study of provable type isomorphisms, and to the decidability of equality in a typed λ-calculus with subtyping.

Capsule Review

This paper investigates a problem in the area of the typed lambda-calculus that comes as a bit of a surprise. The addition of a terminal object causes all approaches to prove strong normalization via reducibility methods to fail. Lambek and Scott show that the extension of the typed lambda-calculus, where the only term of terminal type is the term *, is confluent and SN. The authors extend these results to the plymorphic lambda-calculus, but at the price of having to reduce terms of terminal type to their standard representative first. Although the results of the paper are not the expected ones, namely SN for the whole system, they are important because they imply decidability of the corresponding equational theories and show the limitation of the reducability method.

1 Introduction

Since 1972 there has been some interest in the properties of λ-calculus extended with products and *surjective pairing* (SP), which led to Klop's (1980) discovery that for pure lambda calculus this extension, which we will denote $\lambda^1\beta\eta\pi$, fails to maintain confluence[1], while it remains unproblematic (Pottinger, 1981) for the typed calculus. Due to the connection with Cartesian Closed Categories (CCCs), another extension of the typed calculus has been considered: $\lambda^1\beta\eta\pi*$, which is $\lambda^1\beta\eta\pi$ with *terminal object*. This calculus is relevant for the decision problem of the equational theory of CCCs and for the coherence problem for the same categories, which are discussed in Lambek and Scott (1986) and Mints (n.d.), respectively. Neither of these works provides a truly confluent reduction system for the full calculus: the former takes advantage of type isomorphisms to 'eliminate" the terminal object and reduces the full decision problem to the decision problem for $\lambda^1\beta\eta\pi$ only, the latter gives a system that is Church–Rosser only up to a congruence.

More recent is the interest in $\lambda^1\beta\eta*$, the calculus extended with a terminal object only and no products, which arose in the study of the theory of object oriented programming. In the framework of inheritance, the terminal type **T** has an additional flavour: it is a maximum type. Type inclusion is *not* invariant under isomorphisms, so that, say $A \times \mathbf{T}$ is a type greater than $A \times A'$ for any A', while the same is not true of A alone[2].

Thus the method of solving word problems by first getting rid of the terminal object as in Lambeck and Scott (1986) is of no use in the syntactic theory of λ-calculi with subtyping. We rather need a confluent system for the full type system, terminal (or maximum) type included.

In this paper we exhibit confluent and effectively weakly normalizing (thus decidable) rewriting systems for the full equational theory underlying cartesian closed categories, and for polymorphic extensions of it, bringing the usual interpretation of the extensional equalities η and SP as *contractions* to its extreme limits. To the best of our knowledge, this work provides the first solution to the decidability problem of the full equational theory of Cartesian Closed Categories extended with polymorphic types. Moreover we can take profit of confluence to get conservativity results in addition to decision results. Such conservativity results are needed in the study of provable type isomorphisms.

The results are applied in two companion papers:

- Curien and Ghelli (1990) establish a decidability result in the paradigmatic language $F_{\leq}$, a variant of second-order λ-calculus with a maximum type and bounded quantification: the equational theory considered consists of β, η (first and second-order) and the terminal type rule. We show the confluence and decidability of our system via a translation to the polymorphic λ-calculus with a terminal type (what is called hereafter $\lambda^2\beta\eta*$), and by using a general criterion allowing to transfer confluence in $\lambda^2\beta\eta*$ back to our source system.
- Bruce *et al.* (1992) and Di Cosmo (1994) give an equational characterization of all type isomorphisms which are provable in the typed λ-calculus (respectively

[1] See Barendregt (1984, pp. 403–409), for a short history and references.
[2] L. Cardelli has proposed the following nice and simple exploitation of **T** as a maximum type: consider the well-known inheritance [age;sex] less than [age]; encode [age] as $age \times \mathbf{T}$ and [age;sex] as $age \times (sex \times \mathbf{T})$. Then the desired subtyping obviously holds componentwise, by reflexivity and maximality, respectively.

second order λ-calculus) with pairs and terminal object (what is called hereafter $\lambda^1\beta\eta\pi*$, respectively $\lambda^2\beta\eta\pi*$). It turns out that this characterization can be given quite easily if we are able to determine the structure of *invertible* terms, i.e. terms that possess an inverse w.r.t. the usual operation $\lambda x.\lambda y.\lambda z.(x(yz))$ of composition. The conservativity of equality in the extended calculus over the calculus without products and terminal objects allows us to reduce the problem to the invertibility in the simply typed (respectively second-order) λ-calculus[3].

Technically, we had to navigate between several pitfalls before we arrived to our solution. We survey the main steps of this eventually safe trip in the next section. Sections 3 and 4 are devoted to confluence and weak normalization, respectively. In section 5 we state the decidability and conservativity results that follow quite obviously from confluence and weak normalization, and we put our work in perspective with the other approaches to decidability of the same theories that we are aware of.

2 Survey

After defining precisely the calculi we focus on, we use the Knuth-Bendix procedure by hand to obtain locally confluent rewriting systems. We then shortly hint at a severe technical difficulty in adapting the standard strong normalization proofs which use the so called reducibility method. They can be adapted to a subsystem only. From the confluence of this subsystem we get confluence of almost the whole system by a general criterion presenting an interest of its own. At this stage, only the second-order β-rule is left out, and it can be finally added with the help of Hindley-Rosen's Lemma. As for weak normalization, the ingredients developed for confluence give it for free for first-order systems, while for the second order systems another splitting in subsystems, and another adaptation of the standard strong normalization proofs are needed.

We give now the full definition of the calculus $\lambda^2\beta\eta\pi*$, the most complex of the four we consider.

2.1 The calculus $\lambda^2\beta\eta\pi*$

Definition 2.1
$\lambda^2\beta\eta\pi*$ is the extension of second order lambda calculus defined as follows:

- Types are defined by the following grammar:
  $Type ::= At \mid Var \mid Type \rightarrow Type \mid Type \times Type \mid \forall X.Type$
  where At are countably many atomic types including a distinguished constant type T and Var countably many type variables
- Terms ($M : A$ will stand for *M is a term of type A*)
  — the set of terms contains countably many variables $x, y, \ldots$ of each type

[3] Ultimately the problem is reduced to the invertibility in the untyped λ-calculus (see Barendregt (1984, section 21.2)), where invertible terms have a simple (but not easy to prove!) syntactic characterization due originally to Mariangiola Dezani (Dezani-Ciancaglini, 1976).

— $* : \mathbf{T}$
— if x is a variable of type A and $M : B$, then $\lambda x.M : A \to B$
— if $M : A \to B$ and $N : A$, then $(MN) : B$
— if $M : A$ and $N : B$ then $\langle M, N \rangle : A \times B$
— if $M : A \times B$ then $p_1 M : A$ and $p_2 M : B$
— if $M : A$ and X is a type variable not free in the type of any free variable of M, then $\Lambda X.M : \forall X.A$
— if $M : \forall X.A$ and B is a type, then $M[B] : A[B/X]$.

Notice that pairing and projections are new *term formation rules* and not constants added to the language.

- Equality

$$
\begin{array}{llll}
(\beta) & (\lambda x.M)N = M[N/x] & (\eta) & \lambda x.Mx = M \text{ if } x \notin FV(M) \\
(\pi) & p_i\langle M_1, M_2\rangle = M_i & (\mathbf{SP}) & \langle p_1 M, p_2 M\rangle = M \\
 & & (\mathbf{top}) & M = * \ \text{ if } M : \mathbf{T} \\
(\beta^2) & (\Lambda X.M)[A] = M[A/X] & (\eta^2) & \Lambda X.M[X] = M \text{ if } X \text{ is not free in } M
\end{array}
$$

We will denote $=_{\beta^2\eta^2\pi*}$ the theory of equality generated by β, η, π, SP, *top*, $\beta^2$and η^2.

The other calculi we are interested in can be naturally defined as restrictions of $\lambda^2\beta\eta\pi*$: to obtain them we reduce the class of types and/or terms, and accordingly redefine the equality. The calculus $\lambda^2\beta\eta*$ is $\lambda^2\beta\eta\pi*$ without product types, pairing and projections. (Equality for $\lambda^2\beta\eta*$ will be denoted $=_{\beta^2\eta^2*}$ and is generated by β, η, *top*, β^2 and η^2). The calculus $\lambda^1\beta\eta\pi*$ is $\lambda^2\beta\eta\pi*$ restricted to the first order. (Equality for $\lambda^1\beta\eta\pi*$ will be denoted $=_{\beta\eta\pi*}$ and is generated by β, η, π, SP and *top*). The calculus $\lambda^1\beta\eta*$ is the restriction of $\lambda^1\beta\eta\pi*$ obtained by removing product types, pairing and projections. (Equality for $\lambda^1\beta\eta*$ will be denoted $=_{\beta\eta*}$ and is generated by β, η and *top*).

2.2 *Weakly confluent reduction*

We will adopt the following

Notation 2.2
(Reductions) As usual, $\to$ will denote one-step reduction, while $\to_=$ is the reflexive closure of $\to$, and $\twoheadrightarrow$ is the reflexive transitive closure of $\to$. If the system we consider is weakly normalizing, we will denote $\twoheadrightarrow\mid$ the reduction to a normal form. Also, WN will stand for weakly normalizing, SN for strongly normalizing, CR for confluent (or Church–Rosser) and WCR for weakly (or locally) confluent.

The systems obtained by orienting the equalities of $=_{\beta^2\eta^2\pi*}$ and its restrictions are far from being even weakly confluent, due to a bad interaction between the rule *top* on one side and the rules η and SP on the other[4]. The point is that all terms of

[4] This observation seems to have been first made by A. Obtulowicz (cf. Lambek and Scott (1986, exercise at page 88)).

type **T** are identified (in particular, x:**T** and $*$ are identical), so that λx:**T**.Mx and λx:**T**.M$*$ are 'the same' term, and must give rise to the same reductions: since the first reduces to M, the second must reduce to M too. This fact actually shows up during the completion procedure. Let us consider the typical critical pairs which arise, say, for $\lambda^2\beta\eta\pi*$: after the first 'stage' we find the situation described in figure 1.

	M	M'	M''	New reduction from completion
eta – like	$\lambda x : \mathbf{T}.Mx$	M	$\lambda x : \mathbf{T}.M*$	$\lambda x : \mathbf{T}.M* \longrightarrow M$ if $x \notin FV(M)$
	$\langle p_1 M, p_2 M\rangle$	M	$\langle p_1 M, *\rangle$	$\langle p_1 M, *\rangle \longrightarrow M$ if $M : A \times \mathbf{T}$
	$\langle p_1 M, p_2 M\rangle$	M	$\langle *, p_2 M\rangle$	$\langle *, p_2 M\rangle \longrightarrow M$ if $M : \mathbf{T} \times B$
top – like	$\lambda x : A.Mx$	M	$\lambda x : A.*$	$M \longrightarrow \lambda x : A.*$ if $M : A \to \mathbf{T}$
	$\Lambda X.M[X]$	M	$\Lambda X.*$	$M \longrightarrow \Lambda X.*$ if $M : \forall X.\mathbf{T}$

Fig. 1. The critical pairs at the first stage of Knuth-Bendix completion (M' is reached via η or SP; M'' via *top*).

The additional rules generated by completion can be divided in two groups: rules that behave like η (*eta-like*) and rules that behave like *top* (*top-like*). The former mimick the behaviour of η and SP rules on terms that are known to be 'the same terms as' η and SP redexes, as in the example we just considered above. The latter force to identify all the terms of type $A \to \mathbf{T}$ and $\forall A.\mathbf{T}$, and do pick up a canonical representative in the respective types. It turns out that a set of *eta-like* rules must be generated for each of all types isomorphic (in the categorical sense, see Bruce *et al.* (1992) and Di Cosmo (1994)) to **T**. At stage n, the completion procedure on one side creates new rules to mimick η and SP on terms that are known to be 'the same' as *eta-like* stage $n-1$ redexes, and on the other side it discovers new 'same' terms, following the pattern:

- if A is known to be isomorphic to **T** at stage $n-1$, then $B \to A$ and $\forall X.A$ are isomorphic to **T** at stage n.
- if A and B are known to be isomorphic to **T** at stage $n-2$, then $A \times B$ is isomorphic to **T** at stage n.

These correspond to the well known isomorphisms $\mathbf{T} \times \mathbf{T} \cong \mathbf{T}$, $A \to \mathbf{T} \cong \mathbf{T}$ and $\forall X.\mathbf{T} \cong \mathbf{T}$. (The isomorphism $\mathbf{T} \times \mathbf{T} \cong \mathbf{T}$ shows up only from the second stage on: consider the stage 1 *eta-like* redex $\langle *, p_2 M\rangle$, and suppose $M : \mathbf{T} \times \mathbf{T}$. Then we reach M by the *eta-like* reduction, and $\langle *, *\rangle$ by *top*.)

The following notation will allow us to present in a compact formalism the resulting weakly confluent reduction system.

Definition 2.3
Terminal types and Canonical terms.

1. $Iso(\mathbf{T})$ (the collection of types isomorphic to $\mathbf{T}$) is the set defined as follows:
   (a) $\mathbf{T} \in Iso(\mathbf{T})$
   (b) if $B \in Iso(\mathbf{T})$, then $A \to B \in Iso(\mathbf{T})$ for every type A
   (c) if $A \in Iso(\mathbf{T})$ and $B \in Iso(\mathbf{T})$, then $A \times B \in Iso(\mathbf{T})$
   (d) if $A \in Iso(\mathbf{T})$ and X is a type variable, then $\forall X.A \in Iso(\mathbf{T})$.
2. for each type $A \in Iso(\mathbf{T})$, the associated *canonical* representative $rep(\mathrm{A})$ is defined inductively as follows:

$$\begin{array}{llll} (a) & rep(\mathbf{T}) \text{ is } * & (c) & rep(A \times B) \text{ is } \langle rep(A), rep(B)\rangle \\ (b) & rep(A \to B) \text{ is } \lambda x : A.rep(B) & (d) & rep(\forall X.A) \text{ is } \Lambda X.rep(A). \end{array}$$

Definition 2.4
$\xrightarrow{\beta^2\eta^2\pi*}$ is the notion of reduction for $\lambda^2\beta\eta\pi*$ generated by orienting to the right the equalities β, η, π, SP, $\beta^2$and η^2in Definition 2.1 and adding the following rewriting rules, coming from completion:

$$\begin{array}{ll} (gentop) & M : A \xrightarrow{\beta^2\eta^2\pi*} rep(A) \text{ if } M : A \text{ and } A \in Iso(\mathbf{T}) \text{ and } M \text{ is not already } rep(\mathrm{A}) \\ (SP_{top}) & \langle rep(A), p_2M\rangle \xrightarrow{\beta^2\eta^2\pi*} M \text{ if } M : A \times B \\ (SP_{top}) & \langle p_1M, rep(B)\rangle \xrightarrow{\beta^2\eta^2\pi*} M \text{ if } M : A \times B \\ (\eta_{top}) & \lambda x : A.Mrep(A) \xrightarrow{\beta^2\eta^2\pi*} M \text{ if } A \in Iso(\mathbf{T}) \text{ and } x \notin FV(M). \end{array}$$

The notions of reduction for the simpler calculi can be defined as restrictions of $\xrightarrow{\beta^2\eta^2\pi*}$. The notion of reduction for $\lambda^2\beta\eta*$, which we will denote $\xrightarrow{\beta^2\eta^2*}$, is the reduction induced on $\lambda^2\beta\eta*$ by $\xrightarrow{\beta^2\eta^2\pi*}$, that is to say $\xrightarrow{\beta^2\eta^2\pi*}$ without π, SP, and SP_{top}, as these rules cannot apply to terms of $\lambda^2\beta\eta*$. For the same reason, the clauses for product types in Definition 2.3 will never be used, so that actually only a restricted version of *gentop* is used in $\xrightarrow{\beta^2\eta^2*}$. We shall still use *gentop* to name this restricted reduction, as the intended meaning will always be clear from the context. Similarly, $\xrightarrow{\beta\eta\pi*}$ and $\xrightarrow{\beta\eta*}$ are the reductions induced by $\xrightarrow{\beta^2\eta^2\pi*}$on $\lambda^1\beta\eta\pi*$ and $\lambda^1\beta\eta*$, with the appropriate restrictions of *gentop*.

It is now just a matter of an easy structural induction on terms to see that

Proposition 2.5
$\xrightarrow{\beta^2\eta^2\pi*}$ is weakly confluent (WCR).

What about confluence then? We cannot use the standard Tait-Martin Löf 'parallel reduction' technique, as the non-linear rule SP may require more than one adjustement step, which cannot be parallelized. Specifically, suppose that M one step reduces to M': then $\langle p_1M, p_2M\rangle$ reduces both to M and to $\langle p_1M', p_2M\rangle$. The local confluence diagram can be completed on one side in one step to M', but on the other side one must go sequentially to $\langle p_1 M', p_2 M' \rangle$, where the lost SP redex is recreated, and then to M': this is hardly parallel.

2.3 Investigating strong normalization

Another 'obvious' approach to prove confluence is to attempt to show that these notions of reduction are strongly normalizing, as then one could apply the well known fact that $SN + WCR \Rightarrow CR$ [5]. But here we face a serious problem: some of the new reduction rules, namely η_{top} and SP_{top}, prevent us from applying the usual reducibility techniques (see Girard *et al.*, 1990; Lambek and Scott, 1986; Tait, 1967),), as we briefly sketch now.

All variations of the reducibility method require at some point to show a key statement like *if* $v[u/x] \in RED_V$ *for all* $u \in RED_U$, *then* $\lambda x.v \in RED_{U \to V}$, where RED_T is the set of reducible terms of type T, and where $RED_{U \to V}$ is the set of $s : U \to V$ s.t. $(su) \in RED_V$ for all $u \in RED_U$.

An auxiliary property which is available is that, for $(st) : T$, one has $(st) \in RED_T$ as soon as $s' \in RED_T$ for all s' which are one step reducts of (st).

So the proof of the key statement reduces to the proof that all one step reducts of $(\lambda x.v)\ u$ are reducible. Now, if v is $(v'*)$, then $(\lambda x.v)u$ can reduce to $(v'u)$ which is *not* $v[u/x] = v$, and we do not know if $(v'u)$ is reducible: this does not follow from any of the hypotheses we have at hand. A similar situation arises for SP_{top} when considering the corresponding lemma for pairs. (See the Remark A.14 in Appendix A.)

But the difficulty suggests a solution. The above example is problematic only if u is different from $*$, and this cannot happen if we restrict our attention to terms in *gentop* normal form (*gentop* n.f.). For this to work out we have to check that *gentop* normal forms are stable under reduction. Otherwise the problem could dynamically show up later in the reduction. Unfortunately the β^2 rule does not preserve *gentop* normal forms:

Example 2.6
The second order term $(\Lambda X.\lambda x : X.\lambda y : (X \to A).yx)[\mathbf{T}]$ *is in gentop normal form, but its contractum* $\lambda x : \mathbf{T}.\lambda y : \mathbf{T} \to A.yx$ *is not, and reduces to* $\lambda x : \mathbf{T}.\lambda y : \mathbf{T} \to A.y*$.

So we are forced to drop β^2. Summarizing, so far we have hopes for confluence in the system which is restricted in two ways: we work only with *gentop* normal forms and we have abandoned β^2. Indeed we show that this restricted system is strongly normalizing (Appendix A), thus confluent (the proof of local confluence is easily adapted to the subsystem). Then we lift the confluence result to the system $\xrightarrow{\beta\eta^2\pi*}$, as we will denote the notion of reduction induced on $\lambda^2\beta\eta\pi*$ by $\xrightarrow{\beta^2\eta^2\pi*}$ less β^2 (see next subsection).

Finally we add up β^2, which forms a confluent system that commutes with $\xrightarrow{\beta\eta^2\pi*}$.

[5] Known as *Newman's Lemma*. See Barendregt (1984, p. 58).

So at last we can use Hindley-Rosen's Lemma[6], and we get confluence for the full system $\overset{\beta^2\eta^2\pi*}{\longrightarrow}$.

2.4 *A general criterion for confluence*

To get the confluence of $\overset{\beta\eta^2\pi*}{\longrightarrow}$ from the confluence of its restriction to *gentop* normal forms, we apply the following general method. Recall that two reduction systems R and S are said to commute when, for every term P, if $P \overset{R}{\twoheadrightarrow} Q$ and $P \overset{S}{\twoheadrightarrow} Q'$, there exists a term Q'' such that $Q \overset{S}{\twoheadrightarrow} Q''$ and $Q' \overset{R}{\twoheadrightarrow} Q''$.

Lemma 2.7
Let R be a reduction system that can be split in two subsystems R1 and R2 s.t.

1. R1 is weakly normalizing
2. the set of R1 normal forms is closed w.r.t R2 reductions
3. R2 is confluent on R1 normal forms
4. $\overset{R}{\twoheadrightarrow}$ commutes with $\twoheadrightarrow |\mathrm{R1}$ (see notation 2.2).

Then R is confluent.

Proof
Under the hypothesis above, any two reductions $\overset{R}{\twoheadrightarrow}$ starting from the same term can be completed to the commuting diagram shown in figure 2:

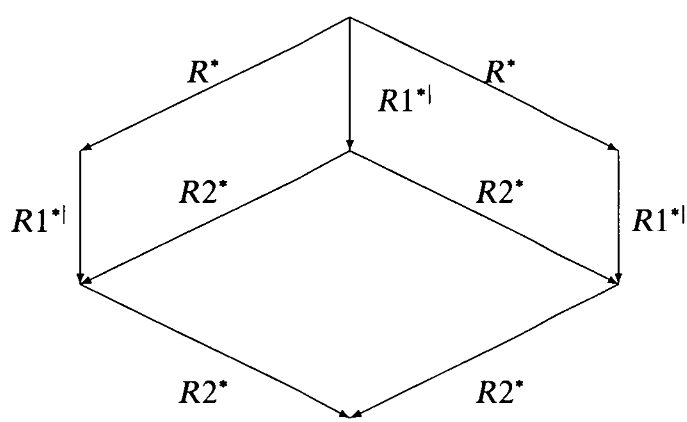

Fig. 2. The factorization of confluence.

- (1) ensures the existence of the R1 normal forms, hence we can build the central vertical arrow in the diagram (R1*| denotes reduction to some R1 n.f.).
- (4) ensures the existence and commutation of the upper inner rhombuses.

[6] The Hindley-Rosen's Lemma asserts the obvious but useful property that two separately confluent, commuting subsystems form a confluent system.

https://doi.org/10.1017/S0956796800001696 Published online by Cambridge University Press

- (2) shows that the lower diagonal arrows in the upper rhombuses are made up of R2 reductions on R1 n.f.'s only, so that (3) guarantees the commutation of the lower inner rhombus.

Finally, the commutativity of the outermost rhombus follows from the commutativity of the inner rhombuses. □

Val Breazu-Tannen pointed out to us that he used a particular case of this very same technique in Breazu-Tannen (1988), to prove Theorem 2.3, even if it was not singled out as a general tool for confluence like here. Later, Val Breazu-Tannen and Jean Gallier (1994) generalized this Theorem to polymorphic lambda calculus, and there too Theorem 4.3 is clearly a particular instance of this technique. This independent discovery and use of this simple tool stresses in our opinion its usefulness.

This criterion is very similar to the *interpretation method* used by Hardin (1989) in her investigations of confluence properties of categorical combinators, even if neither one is an instance of the other.

Our travel is close to the end. We shall take $\xrightarrow{\beta\eta^2\pi*}$ as R, *gentop* as R1, $\xrightarrow{\beta\eta^2\pi*}$ less *gentop* as R2 and prove the four conditions of the criterion. The confluence of R2 on R1 normal forms is proved by establishing WCR and SN.

3 Confluence

Let in the following R stand for one of $\xrightarrow{\beta\eta^2\pi*}$, $\xrightarrow{\beta\eta^2*}$, $\xrightarrow{\beta\eta\pi*}$ or $\xrightarrow{\beta\eta*}$, R1 be *gentop* and R2 be R less *gentop*. It will be intended that in the case of $\xrightarrow{\beta\eta\pi*}$ and $\xrightarrow{\beta\eta*}$, we consider only first order terms and types and hence only the corresponding restricted form of *gentop*, for which the following proofs hold almost unchanged.

We first introduce some notation.

Notation 3.1
We will denote $(M)^T$ the *gentop* n.f. of a term M and $\overset{gentop}{\twoheadrightarrow}$ |the reduction to *gentop* normal form $(M)^T$. Notice that throughout the paper we will use the $=$ sign to mean 'identical up-to α-conversion'.

Lemma 3.2
The following equalities hold:

1. $(PQ)^T = (P)^T(Q)^T$ if (PQ):A and $A \notin Iso(\mathbf{T})$
2. $(p_iP)^T = p_i(P)^T$ if p_iP:A and $A \notin Iso(\mathbf{T})$
3. $(\lambda x.P)^T = \lambda \mathrm{x}.(P)^T$
4. $(\langle P,Q\rangle)^T = \langle (P)^T,(Q)^T\rangle$
5. $(\Lambda X.P)^T = \Lambda X.(P)^T$
6. $(P[B])^T = (P)^T[B]$ if $P[B]{:}A \notin Iso(\mathbf{T})$.

Proof
We only check 3, and leave the rest to the reader. Let $\lambda x.P : A \to B$. If $A \to B \notin Iso(\mathbf{T})$, then the result is trivial, otherwise $(\lambda x.P)^T = rep(A \to B) = \lambda z.(P)^T$ for some fresh variable z. Since $(P)^T$, which is equal to rep(B), has no occurrence of variables in it (easily shown by induction), then $\lambda z.(P)^T$ is equal to $\lambda x.(P)^T$ by α-conversion.
□

Table 1. *Compatibility of gentop n.f. with substitution.*

M	LHS	RHS	Comment
x	$(N)^T$	$(N)^T$	ok
y	y	y	ok
(PQ)	$((PQ)[N/x])^T$	$(PQ)^T[(N)^T/x]$	def. subst., 3.2
	$=(P[N/x]Q[N/x])^T$	$= ((P)^T(Q)^T)[(N)^T/x]$	3.2, def. subst.
	$= ((P[N/x])^T(Q[N/x])^T)$	$= ((P)^T[(N)^T/x](Q)^T[(N)^T/x])$	ind. hyp.
	$= ((P)^T[(N)^T/x])((Q)^T[(N)^T/x])$		
λy.P	$(\lambda y.P[N/x])^T$	$(\lambda y.P)^T[(N)^T/x]$	3.2, 3.2
	$= \lambda y.(P[N/x])^T$	$= (\lambda y.(P)^T)[(N)^T/x]$	ind. hyp., def. subst.
	$= \lambda y.(P)^T[(N)^T/x]$	$= \lambda y.(P)^T[(N)^T/x]$	
p_iP	$(p_iP[N/x])^T$	$(p_iP)^T[(N)^T/x]$	3.2, 3.2
	$= p_i(P[N/x])^T$	$= (p_i(P)^T)[(N)^T/x]$	ind. hyp., def. subst.
	$= p_i(P)^T[(N)^T/x]$	$= p_i(P)^T[(N)^T/x]$	
$\langle P,Q\rangle$	$.(\langle P[N/x],Q[N/x]\rangle)^T$	$(\langle P,Q\rangle)^T[(N)^T/x]$	3.2, 3.2
	$= \langle (P[N/x])^T,(Q[N/x])^T\rangle$	$= \langle (P)^T,(Q)^T\rangle[(N)^T/x]$	ind. hyp., def. subst.
	$= \langle (P)^T[(N)^T/x],(Q)^T[(N)^T/x]\rangle$	$= \langle (P)^T[(N)^T/x],(Q)^T[(N)^T/x]\rangle$	
Λt.P	$(\Lambda t.P[N/x])^T$	$(\Lambda t.P)^T[(N)^T/x]$	3.2, 3.2
	$= \Lambda t.(P[N/x])^T$	$= (\Lambda t.(P)^T)[(N)^T/x]$	ind. hyp., def. subst.
	$= \Lambda t.(P)^T[(N)^T/x]$	$= \Lambda t.(P)^T[(N)^T/x]$	
$P[A]$	$(P[A][N/x])^T$	$(P[A])^T[(N)^T/x]$	def. subst., 3.2
	$= (P[N/x][A])^T$	$= (P)^T[A][(N)^T/x]$	3.2, def. subst.
	$= (P[N/x])^T[A]$	$= (P)^T[(N)^T/x][A]$	ind. hyp.
	$= (P)^T[(N)^T/x]A[]$		

Lemma 3.3
$\overset{gentop}{\twoheadrightarrow}$ |is compatible with substitution, i.e.

$$(M[N/x])^T = (M)^T[(N)^T/x]$$

Proof
By an easy induction on the structure of M (see Table 1). Notice that the case $M : U$ and $U \in Iso(\mathbf{T})$ is trivial since in both cases the normal form is *rep*(U), so in the table we consider only the case when the normal form of a compound term is the combination of the normal forms of its components. ☐

Lemma 3.4
If $M \xrightarrow{R} M'$ then $(M)^T \xrightarrow{R}_= (M')^T$.

Proof
We will proceed by induction on the structure of M. Notice that whenever M is a *gentop* redex, the claim holds trivially since the reductions we consider all preserve the type of the redex: so the type of M' is the same as that of M and their *gentop* normal forms are the same[7]. We shall thus assume that M is not a *gentop* redex. Furthermore, if the R reduction takes place in a proper subterm of M, the result

[7] Remember that the contractum of a *gentop* redex depends only on the type of the redex, not on its structure.

https://doi.org/10.1017/S0956796800001696 Published online by Cambridge University Press

follows easily by induction in each case (by Lemma 3.2), so we will not state it explicitly. We are left with the hypothesis that M is a redex which is not a *gentop* redex.

- M is a variable x. No reduction is possible, and the statement holds vacuously.
- M is an application. There is only one case:
  - — M is $(\lambda x.P')Q$ and it β reduces to $P'[Q/x]$. Then $(M)^T = ((\lambda x.P')^T(Q)^T) = (\lambda x.(P')^T)(Q)^T$, and it β reduces to $(P')^T[(Q)^T/x]$, which is equal to $(P'[Q/x])^T$ by compatibility of $\overset{gentop}{\twoheadrightarrow}$ |with substitution (Lemma 3.3).
- M is an abstraction. There are two cases:
  - — M is $\lambda x.(Px)$ and it η reduces to P. Then we have two possibilities for $(M)^T$ (notice that $(Px)^T = rep(\mathrm{V})$ is excluded as then M would be a *gentop* redex):
  - — $\lambda x.((P)^T x)$ which η reduces to $(P)^T$
  - — $\lambda x.((P)^T rep(U))$ which η_{top} reduces to $(P)^T$
- M is $\lambda x.(Prep(U))$ and it η_{top} reduces to P. Then $(M)^T = \lambda x.((P)^T rep(U))$ which η_{top} reduces to $(P)^T$.
- M is a projection.The only case to consider is
  - — M is $p_i\langle P_1, P_2\rangle$ and it π reduces to P_i. Then $(M)^T$ is $p_i(\langle P_1, P_2\rangle)^T$, which is $p_i\langle (P_1)^T, (P_2)^T\rangle$, which π reduces to $(P_i)^T$.
- M is a pair. There are three cases:
  - — M is $\langle p_1P, p_2P\rangle$ and it SP reduces to P. By lemma 3.2, we focus only on the following three possibilities for $(M)^T$:
    - $\langle p_1(P)^T, p_2(P)^T\rangle$ which SP reduces to $(P)^T$
    - $\langle p_1(P)^T, rep(V)\rangle$ which SP_{top} reduces to $(P)^T$
    - $\langle rep(U), p_2(P)^T\rangle$ which SP_{top} reduces to $(P)^T$
  - — M is $\langle p_1P, rep(V)\rangle$ and it SP_{top} reduces to P.
    Then $(M)^T$ is $\langle p_1(P)^T, rep(V)\rangle$ which SP_{top} reduces to $(P)^T$
  - — M is $\langle rep(U), p_2P\rangle$ and it SP_{top} reduces to P.
    Then $(M)^T$ is $\langle rep(U), p_2(P)^T\rangle$ which SP_{top} reduces to $(P)^T$.
- M is an abstraction $\Lambda t.P$. There is only one case to consider, namely P is $P'[X]$ and reduces to P' via η^2. We can assume $P'[X]$ not to be a *gentop* redex, as otherwise $M = \Lambda X.P'[X]$ would be a *gentop* redex too, while we already factored out the case $M{:}U \in Iso(\mathbf{T})$. By Lemma 3.2, $(M)^T = (\Lambda X.P'[X])^T = \Lambda X.(P'[X])^T = \Lambda X.(P')^T[X]$, which reduces via η^2 to $(P')^T$, as required.

Hence we have shown that $(M)^T \xrightarrow{R}_= (M')^T$. □

Using the criterion for confluence, we will now show

Theorem 3.5
R is confluent.

https://doi.org/10.1017/S0956796800001696 Published online by Cambridge University Press

Proof

We check the four hypotheses of lemma 2.7 for R split in R1 and R2 as above.

1. *gentop is a strongly normalizing confluent reduction system.*
   Each *gentop* step strictly decreases the number of *gentop* redexes in the term it is applied to. Since it is also trivially WCR, Newman's Lemma applies and we get CR too.
2. *R2 reductions do not create new gentop redexes.*
   By cases on the rule which is used. For all rules but β the result obviously follows from the fact that the reduct is a subterm of the redex. The case β is settled by noticing that, if M and N are in *gentop* n.f., then $M[N/x]$ is in *gentop* n.f. too. Indeed, this last property can be easily shown by induction on the structure of M.
   If M is x or if it does not contain x free, then $M[N/x]$ is either M or N and the result follows from the hypothesis. We can also rule out the case where M is $rep(\mathrm{A})$, as then it has no free variables. So $M : A \notin Iso(\mathbf{T})$. If $M[N/x]$ contains a *gentop* redex P, then P cannot be $M[N/x]$, which has the same type as M, so P must be a proper subterm of $M[N/x]$. P cannot be a subterm of N either, or an unchanged subterm of M, as they are already in normal form, so it must be $M'[N/x]$ with M' a proper subterm of M containing a free occurrence of x. But M' is in *gentop* normal form as M is, hence, by induction hypothesis $M'[N/x]$ is not a *gentop* redex, so $M[N/x]$ is in *gentop* n.f.
3. *The systems* $\xrightarrow{\beta\eta^2\pi*}$, $\xrightarrow{\beta\eta^2*}$, $\xrightarrow{\beta\eta\pi*}$ *and* $\xrightarrow{\beta\eta*}$ *are confluent over gentop normal forms.*
   All the systems introduced so far are weakly confluent. We will prove in the appendix (theorem A.19, which follows closely the proof plan of Girard *et al.* (1990)), that $\xrightarrow{\beta\eta^2\pi*}$ is strongly normalizing over *gentop* normal forms. This implies strong normalization (over *gentop* normal forms) for all the others subsystems of it. Hence they are confluent over *gentop* n.f.'s by Newman's Lemma.
4. *If* $M \overset{R}{\twoheadrightarrow} M'$ *then for any gentop n.f.* N *of* M *and* N' *of* M' $N \overset{R}{\twoheadrightarrow} N'$.
   By Lemma 3.4 above and a simple diagram chase.

We can finally conclude, by lemma 2.7, that R is confluent. □

Remark 3.6

Statement 4 of the previous theorem holds for all the reduction systems we are considering, as we showed it for $\xrightarrow{\beta\eta^2\pi*}$, and the statements for the other ones are particular cases of it.

Corollary 3.7

R2 is confluent on *gentop* n.f.'s

Proof

Statement 3 of the previous theorem tells us that if $M \overset{R}{\twoheadrightarrow} M'$ and $M \overset{R}{\twoheadrightarrow} M''$, where M is in *gentop* normal form, then we can find M''' s.t. $M' \overset{R}{\twoheadrightarrow} M'''$ and $M'' \overset{R}{\twoheadrightarrow} M'''$. Now the second point shows that any reduction path starting from a *gentop* n.f.

cannot contain *gentop* reductions, so the R reductions are made up only of R2 steps and we get the result. □

We still have a gap to fill for the second-order systems, since we have left out β^2. We shall prove CR for $\xrightarrow{\beta^2\eta^2\pi*}$ and $\xrightarrow{\beta^2\eta^2*}$ by using Hindley-Rosen's Lemma.

Let R1 be the system $\xrightarrow{\beta\eta^2\pi*}$ (or $\xrightarrow{\beta\eta^2*}$) and R2 be β^2.

Lemma 3.8
β^2 is confluent.

Proof
The system consisting of β^2 alone satisfies the diamond property, hence is CR. □

We just proved that R1 is CR (Theorem 3.5), so we are left to show that R1 commutes with R2, and the CR property will follow by Hindley-Rosen's Lemma.

Theorem 3.9
R1 and R2 commute with each other.

Proof
It suffices to prove that, if $M \xrightarrow{R1} M'$ and $M \xrightarrow{R2} N$, then there exist a term M'' s.t. $N \overset{R2}{\twoheadrightarrow} M''$ and $M' \xrightarrow{R1}_{=} M''$ (see Lemma 3.3.6 in (Barendregt, 1984), pag. 65). The only superpositions arise with η^2 and *gentop*, and are easily closed up, so that it suffices to notice that β^2 cannot duplicate existing redexes (β^2 can only duplicate types, that are not redexes), so that the constraint on the R1 reduction which closes the diagram gives no problem. The details are left to the reader. □

So we finally get, by Hindley-Rosen's Lemma.

Theorem 3.10
The systems $\xrightarrow{\beta^2\eta^2\pi*}$ and $\xrightarrow{\beta^2\eta^2*}$ are confluent [8].

4 Weak normalization

For the first order systems, we get from the previous section a normalizing strategy for free: first go to the *gentop* normal form, then use the SN property on *gentop* normal forms.

Summarizing, we have obtained:

Theorem 4.1
The calculi $\lambda^1\beta\eta*$, $\lambda^1\beta\eta\pi*$are effectively weakly normalizing.

[8] We also found an alternative proof of the confluence of $\xrightarrow{\beta^2\eta^2*}$ that does not extend to the case with *SP*. It relies on yet another splitting of the rules, taking *gentop* and the β rules on one hand, and the *eta-like* rules on the other. The proof uses the same criterion for confluence as we used in this section. In order to check the last condition, we rely on a parallelization of R2, which does not work well when the non linear surjective pairing rule is added to R2 (cf. introduction). So we abandoned that proof technique which we were not able to extend to the full system.

Since for the second order systems we have left out $\beta^2$and η^2, we find them on the way: we can deal with them at the price of a splitting of the set of rules which is different from the splitting which lead us to confluence.

Theorem 4.2
The calculi $\lambda^2\beta\eta*$, $\lambda^2\beta\eta\pi*$ are effectively weakly normalizing.

Proof
The reduction system R can be split into the two subsystems R1 $= \{\beta, \pi, gentop, \beta^2, \eta^2\}$ and R2 $= \{\eta, SP, \eta_{top}, SP_{top}\}$. R1 is shown to be SN by a straightforward adaptation of the technique of Girard *et al.* (1990) (see Appendix B). R2 is obviously SN since the rules strictly decrease the size of the terms they apply to.

One can then show by an easy induction on the structure of the context surrounding an R2 redex that no R2 reduction creates any *new* R1 redex. □

Theorem 4.3
R2 reductions do not create new R1 redexes.

Proof
It suffices to consider the case of $\lambda^2\beta\eta\pi*$, as the R1 and R2 systems for it embody the R1 and R2 systems for all the others.
First notice that since R2 reductions preserve the type, no new *gentop* redex can be created as *gentop* redexes depend only on the type of the terms.
As for β, π, β^2 and η^2, let $P \xrightarrow{R2} P'$.
A context with a single hole for our calculus can be defined inductively as follows:

$$C[\,] := [\,] \mid (QC[\,]) \mid (C[\,]Q) \mid p_iC[\,] \mid \lambda x.C[\,] \mid \langle Q, C[\,]\rangle \mid \langle C[\,], Q\rangle \mid \Lambda X.C[\,] \mid C[\,][A]$$

We prove the lemma by induction on the context $C[\,]$ where the R2 redex P occurs. Notice that the only interesting cases are when P appears in a position where a new R1 redex can be created, i.e. when it is applied to a term or it appears in p_iP.

- [] trivial since P' is a subterm of P for all rules in R2
- $(QC[\,])$ by induction hypothesis, $C[P']$ contains no R1 redexes not appearing in $C[P]$. Since the fact that the application $(QC[\,])$ is a redex depends on Q only, which does not change, and redexes inside Q do not change too, we are done.
- $(C[\,]Q)$ by induction hypothesis, $C[P']$ contains no R1 redexes not appearing in $C[P]$. Q does not change, so redexes inside Q do not change too.
  The only possible new redex would be the application $(C[P']Q)$ if $C[P']$ is an abstraction and $C[P]$ is not. This can happen only if $C[P]$ is P, and due to typing reasons, this means $(PQ) \xrightarrow{\eta} (P'Q)$ or $(PQ) \xrightarrow{\eta_{top}} (P'Q)$.
  In both cases P is already an abstraction, so this redex is not new either and we are done.
- $p_iC[\,]$ by induction hypothesis, $C[P']$ contains no R1 redexes not appearing in $C[P]$.
  The only possible new redex would be $p_iC[P']$ if $C[P']$ is a pair and $C[P]$ is

not. Again, this can happen only if $C[P]$ is P, and due to typing reasons, this means $P \xrightarrow{SP} P'$ or $P \xrightarrow{SP_{top}} P'$.
In both cases P is already a pair, so this redex is not new either and we are done.

- $\lambda x.C[\,]$ by induction hypothesis, $C[P']$ contains no R1 redexes not appearing in $C[P]$. Since an abstraction is not an R1 redex, the same holds for $\lambda x.C[P]$.
- $\langle Q, C[\,]\rangle$, $\langle C[\,], Q\rangle$, $\Lambda X.C[\,]$, $C[\,][A]$: similarly as for abstraction.

□

This has the following important consequence

Corollary 4.4
The set of R1 normal form is closed w.r.t. R2 reductions.

Since R2 is obviously SN, as the rules strictly decrease the size of the terms they apply to, this corollary gives us the following, very easy, effective normalizing (standard) strategy.

Given a term M,

1. first R1–normalize it reaching, say, M',
2. then R2–normalize M' reaching, say, M''.

M'' is the desired normal form.

The previous result about weak normalization for the first order fragment can obviously be derived as a corollary from this theorem, but we actually needed the ingredients of the previous proof to get the confluence of our systems.

5 Decidability and conservative extension results

From the confluence and weak normalization for our calculi, it is now easy to get also the decidability of the associated equational theories as well as conservativity results.

Corollary 5.1
The equational theories for $\lambda^1\beta\eta*$, $\lambda^1\beta\eta\pi*$, $\lambda^2\beta\eta*$ and $\lambda^2\beta\eta\pi*$are decidable.

Proof
Given terms M and N, consider their normal forms M' and N' (they exist by WN). If $M = N$, then (by CR) M' is syntactically equal to N'. So, to decide equality it suffices to take the normal forms (which is effective, as we provided a normalizing strategy for each one of these calculi) and to check if they are equal. □

Corollary 5.2
(Conservative extensions) For L any of the calculi $\lambda^2\beta\eta\pi*$, $\lambda^2\beta\eta*$, $\lambda^1\beta\eta\pi*$ or $\lambda^1\beta\eta*$, call $\xrightarrow{L}$ the rewriting system corresponding to L, that is $\xrightarrow{\beta^2\eta^2\pi*}$, $\xrightarrow{\beta^2\eta^2*}$, $\xrightarrow{\beta\eta\pi*}$ or $\xrightarrow{\beta\eta*}$. Let L' be a subtheory of L which has the following stability property. If M is in the sublanguage of L' and $M \xrightarrow{L} N$, then N is also in L' and M and N are provably equal in L'. If M and N are terms of L' that are equal in L, then they are already equal in L'.

https://doi.org/10.1017/S0956796800001696 Published online by Cambridge University Press

Proof
If M and N are equal in L, then, by the CR property, there exist a term P s.t. M and N reduce to P in L. But M and N are terms of L', and no reduction in any of the calculi we consider can reach terms outside L', then the reductions $M \overset{L}{\twoheadrightarrow} P$ and $N \overset{L}{\twoheadrightarrow} P$ correspond to provable equations in L', so that M is equal to N in L'. $\square$

In Bruce *et al.* (1992), for example, we need the conservativity of the equational theory of $\lambda^1\beta\eta\pi*$ over the simple typed λ-calculus, while in Di Cosmo (1994), we actually use the conservativity of $\lambda^2\beta\eta\pi*$ over the second order lambda calculus.

As far as we know, our results are new for what concerns polymorphism, while other proofs of corollary 5.1 have been given in the literature, for the case of the first order calculi. We already briefly hinted at the method used in Lambek and Scott (1986), which is based on

- the elimination of Top
- a proof of confluence via WCR and SN (WCR holds there without a need to add funny rules, and the computability method works well without special restrictions, as was first shown by R. De Vrijer (1987)).

Another method, which was found independently by Troelstra (1986), where it is used to prove SN rather than CR) and Hardin (1989) goes further by eliminating products as well as Top. The two methods allow to prove conservativity as well as decidability, but the overall construction is quite tedious. Let us be more specific, since the explanations provided by Lambek and Scott (1986, pp. 81, 82), are somewhat handwaving. The exploitation of the type isomorphisms can be formalized as follows. To every type T we associate a $\mathbf{T}$-*free* type $T^\diamond$.

Definition 5.3
For any type T, we define its 'top-free' form $T^\diamond$ as the normal form of T w.r.t. the following (confluent and strongly normalizing) type rewrite system $\rightsquigarrow$:

$$
\begin{array}{lll}
A \times \mathbf{T} \rightsquigarrow A & \qquad & \mathbf{T} \times A \rightsquigarrow A \\
\mathbf{T} \rightarrow A \rightsquigarrow A & \qquad & A \rightarrow \mathbf{T} \rightsquigarrow \mathbf{T}
\end{array}
$$

Thus a '$\mathbf{T}$-free' type is either $\mathbf{T}$, or a type where $\mathbf{T}$ does not occur. Then one may extend this mapping to terms, so that for a term $M : A$ we have $M^\diamond : A^\diamond$, in such a way that

$$M =_{\beta\eta\pi*} N \Longleftrightarrow M^\diamond =_{\beta\eta\pi} N^\diamond$$

Similarly, to a type A of $\lambda^1\beta\eta\pi*$ we can associate a sequence of types A^* constructed from type variables with the arrow only, and to a term M a sequence M^* of terms of the types that appear in A^*. Then $M =_{\beta\eta\pi*} N$ iff $M_1 =_{\beta\eta} N_1, \ldots, M_n =_{\beta\eta} N_n$, where $M^* = M_1,\ldots,M_n$ and $N^* = N_1,\ldots,N_n$.

This formalizes the assertion of Lambek and Scott that there is 'no loss of generality', as far as decision is concerned, if one removes the terminal object (or both the terminal object and the products).

Moreover, these translations of types and terms are conservative in the sense that if A is a type where $\mathbf{T}$ (respectively $\mathbf{T}$ and $\times$) does not occur, and $M : A$, then $A^\diamond$

and M° (respectively $A^{\bullet}$ and $M^{\bullet}$) are just A and M. Corollary 5.2 is an immediate consequence of this.

Yet another solution to the decidability problem for equational theories of cartesian closed categories has been proposed by Obtulowicz (1987). His approach is very algebraic in nature. Obtulowicz defines effectively operations on some canonical forms, turning the set of canonical forms into an initial algebra. Then, to decide that two terms are equal, one computes their interpretation in the initial algebra, and checks whether the resulting canonical forms coincide. This approach is very technical, and contains hidden rewriting techniques. But it is interesting, because it does not a priori require such strong assumptions as to find a noetherian and confluent rewriting system.

Anyway, Obtulowicz did not show decidability for exactly the same equational theories as we do here. Specifically, he deals with the critical pairs which lead us to the SP_{top} rules in a different way. He forces an equational theory on types as well as on terms. Specifically, the canonical type isomorphisms underlying the translation $^{\circ}$ above are forced to be true equalities (and models of these theories have thus to identify on the nose, say $A \times \mathbf{T}$ and A). A set E of new equations between terms are added, which witness these identifications at the level of terms. Here is one of them

$$\langle M, * \rangle =_E M \text{ for } M : A \times \mathbf{T}$$

With the aid of this equation and of one of its consequences, namely

$$p_1 M =_E \mathrm{M} \text{ for } \mathrm{M} : A \times \mathbf{T}$$

one can solve the critical pair

$$\langle p_1 M, * \rangle \leftarrow \langle p_1 M, p_2 M \rangle \rightarrow M$$

by just noting that $\langle p_1 M, * \rangle \rightarrow p_1\mathrm{M} \rightarrow \mathrm{M}$. It would be worthwhile to investigate these theories from a rewriting point of view.

Another treatment of the terminal object with identification of types can be found in Nipkow (1990), which is only concerned with *local* confluence.

Let us mention that the problem of finding a confluent completion of the theory $\lambda^1\beta\eta\pi*$ has been considered in Poigné and Voss (1987), where it was believed to be solved. Unfortunately, Poigné and Voss (1987) missed the critical pair leading to η_{top}, which in turn induced them to believe that the adaptation of the standard SN proof was straightforward.

Another interesting approach is based on the idea of turning η and SP into *expansions* instead of contractions, getting a strongly normalizing system at the price of some restrictions on the reductions which take into accout the *context* where a redex occurs. The system so obtained is not a rewrite system in the usual sense, not even a conditional one, due to these contextual constraints that invalidate several usual properties of reductions in the λ-calculus, but has the advantage of using a finite number of rules. This approach was taken in several works (Akama, 1993; Cubric, 1992; Di Cosmo and Kesner, 1994b; Dougherty, 1993; Jay and Ghani, 1992; Di Cosmo and Kesner, 1994a). For a full discussion of this approach, and complete references, we refer the interested reader to Di Cosmo and Kesner (1994b).

Appendix: Strong normalization for subsystems

Our proof of confluence in Theorem 3.5 relies upon the strong normalization of $\stackrel{\beta\eta^2\pi*}{\longrightarrow}$ over the set of *gentop* normal forms, while we need the strong normalization of $\stackrel{\beta^2\eta^2\pi*}{\longrightarrow}$ less η_{top} and SP_{top} over the full set of terms in order to provide an effective weakly normalizing strategy for $\stackrel{\beta^2\eta^2\pi*}{\longrightarrow}$ in Theorem 4.2.

This appendix provides these two proofs of strong normalization in section A and B respectively, by suitably adapting one of the various versions of the reducibility method. We choose here to apply Girard's method, following essentially the same proof plan as in Girard *et al.* (1990, pp. 42–47). Since there is almost no difference in the proofs for the two systems, we will detail the first one only , and only point out the differences for the second case.

As we briefly suggested in the introduction (section 2.3), the reducibility method fails for the full system where η_{top} and SP_{top} are allowed to freely interact with any term of the calculus: we are not able to deal in the crucial proofs of the abstraction and pairing lemmas (Lemmas A.13 and A.12) with some reductions that arise in the full system.

To rule out these reductions, one can either restrict the system to *gentop* normal forms only (this requires in turn to rule out the β^2 rule, that does not preserve *gentop* normal forms, as shown in Example 2.6), or one can simply rule out η_{top} and SP_{top}.

A Normalization without β^2 on *gentop* n.f.s

In this section we will show that the system $\stackrel{\beta\eta^2\pi*}{\longrightarrow}$ (the full system $\stackrel{\beta^2\eta^2\pi*}{\longrightarrow}$ less β^2) is strongly normalizing over the set of *gentop* normal forms. This means that all along the proof *any gentop reduction is ruled out*, so we will not explicitly state all the time that *gentop* reductions cannot occur. Moreover, to improve readability, $\longrightarrow$ will stand for $\stackrel{\beta\eta^2\pi*}{\longrightarrow}$ in this section.

Definitions

Definition A.1 (neutral terms)
A term t:U is neutral iff one of the following conditions is satisfied:

- if $U \notin Iso(\mathbf{T})$ and t is not an abstraction, a type abstraction or a pair, or
- if $U \in Iso(\mathbf{T})$ (then t is $rep(U)$, as we consider only terms in *gentop* normal form).

Definition A.2 (longest reduction path for a term)
For a term u, ν(u) denotes the length of the longest reduction path starting from u. Notice that, by König's Lemma, if u is strongly normalisable, then ν(u) is finite.

Definition A.3
A *reducibility candidate* of type U is a set R of terms of type U with the following properties.

CR1 if $t \in R$, then t is strongly normalisable.
CR2 if $t \in R$ and $t \to t'$, then $t' \in R$.
CR3 if t is neutral and for all t' s.t. $t \to t'$ we have that $t' \in R$, then $t \in R$.

Remark A.4
A reducibility candidate R of type U is never empty:

- If $U \in Iso(\mathbf{T})$, then $rep(U)$ is neutral and in normal form and hence belongs to R by (CR3).
- If $U \notin Iso(\mathbf{T})$, then any variable of type U is neutral and in normal form and hence belongs to R by (CR3).

Proposition A.5
The set of strongly normalizable terms of type U is a reducibility candidate.

Proof
- (CR1) is a tautology.
- (CR2) if t is strongly normalisable, then every t' s.t. t $\longrightarrow$ t' is strongly normalisable.
- (CR3) every reduction path leaving t must pass through one of the terms t' that are one step from t. Since all t' are strongly normalizable, then t is strongly normalisable also.

$\square$

Definition A.6 (product and arrow of reducibility candidates)
If R and S are reducibility candidates of types U and V, we can define sets $R \to S$ of terms of type $U \to V$ and $R \times S$ of terms of type $U \times V$ as follows:

- $t \in R \to S$ (of type $U \to V$) $\Longleftrightarrow$
  - for all $u \in R$, $(tu) \in S$ if $V \notin Iso(\mathbf{T})$
  - $t = rep(U \to V)$ if $V \in Iso(\mathbf{T})$
- $t \in R \times S$ (of type $U \times V$) $\Longleftrightarrow$
  - $p_1 t \in R$ and $p_2 t \in S$ if U, V are not in $Iso(\mathbf{T})$
  - $p_1 t \in R$ if $U \notin Iso(\mathbf{T})$, $V \in Iso(\mathbf{T})$
  - $p_2 t \in S$ if $U \in Iso(\mathbf{T})$, $V \notin Iso(\mathbf{T})$
  - $t = rep(U \times V)$ if $U, V \in Iso(\mathbf{T})$

Remark A.7
Notice that, as t and u are in *gentop* normal form, and due to the conditions on U and V, the terms (tu), $p_1 t$ and $p_2 t$ above are still in *gentop* normal form.

Theorem A.8
If R_1 and R_2 are reducibility candidates of types U_1 and U_2, then $R_1 \times R_2$ and $R_1 \to R_2$ are reducibility candidates of type $U_1 \times U_2$ and $U_1 \to U_2$ respectively.

Proof
Assume that R_1 and R_2 are reducibility candidates of type U_1 and U_2, respectively.

1. $R_1 \times R_2$ is a reducibility candidate of type $U_1 \times U_2$. If $U_1 \times U_2 \in Iso(\mathbf{T})$, then (CR1), (CR2) and (CR3) hold vacuously due to the fact that we consider only *gentop* normal forms, so let's assume in the following that $U_1 \notin Iso(\mathbf{T})$ and/or $U_2 \notin Iso(\mathbf{T})$.

   - (CR1) if $t \in U_1 \times U_2$ and $U_i \notin Iso(\mathbf{T})$, then p_it is strongly normalisable by the induction hypothesis on U_i, since p_it $\in U_i$ by definition. Hence t is strongly normalisable.
   - (CR2) if $t \longrightarrow t'$, then p_1t $\longrightarrow p_1$t' and/or p_2t $\longrightarrow p_2$t'. As $t \in U_1 \times U_2$, then p_1t $\in U_1$ and/or p_2t $\in U_2$. By induction hypothesis **CR2** for U_1 and/or U_2 we get p_1t' $\in U_1$ and/or p_2t' $\in U_2$ and hence, by definition, $t' \in U_1 \times U_2$.
   - (CR3) t is neutral and all t' one step from t are in $U_1 \times U_2$.
     We need to show p_1t $\in U_1$ and/or p_2t $\in U_2$. Now notice that applying a conversion inside p_it can only result in some p_it' as t is not a pair (it is neutral and it is not $rep(U_1 \times U_2)$). But p_1t' $\in U_1$ and/or p_2t' $\in U_2$ as t' is in $U_1 \times U_2$. In any case, p_1t and/or p_2t are neutral and every term one step from it is in $U_1 \times U_2$, so the induction hypothesis for U_1 and/or U_2 ensure p_1t $\in U_1$ and/or p_2t $\in U_2$. So $t \in U_1 \times U_2$.

2. $R_1 \rightarrow R_2$ is a reducibility candidate of type $U_1 \rightarrow U_2$.
   We can assume that $U_2 \notin Iso(\mathbf{T})$ since otherwise $U_1 \rightarrow U_2 \in Iso(\mathbf{T})$, and then (CR1), (CR2) and (CR3) hold vacuously.

   - (CR1) if $t \in U_1 \rightarrow U_2$, then let u be a variable x of type U_1 if $U_1 \notin Iso(\mathbf{T})$ or else $rep(U_1)$. Since $u \in$ *any* reducibility candidate, (remark A.4), we get that $(tu) \in U_2$ by definition, hence (tu) is strongly normalisable by induction hypothesis for U_2, that suffices to show that t is strongly normalisable.
   - (CR2) if $t \longrightarrow t'$, we need to show $(t'u) \in U_2$ for all $u \in U_1$. Take then $u \in U_1$; we have $(tu) \in U_2$ and $(tu) \longrightarrow (t'u)$, and hence $(t'u) \in U_2$ by induction hypothesis on U_2.
   - (CR3) t is neutral and all t' one step from t are in $R_1 \rightarrow R_2$. In order to show $t \in U_1 \rightarrow U_2$, we need to show $(tu) \in U_2$ for all $u \in U_1$. By induction hypothesis on U_1, we get u is strongly normalisable, so we can argue by induction on ν(u). In one step, (tu) converts to:

     — $(t'u)$ with t' one step from t.
     As $t' \in U_1 \rightarrow U_2$, we get $(t'u) \in U_2$ by definition.

     — (tu') with u' one step from u.
     By induction hypothesis on U_1, $u' \in U_1$ and ν(u') < ν(u), so $(tu') \in U_2$ by the induction hypothesis on u.

     — there is no other possibility, as t is already in *gentop* n.f. and it is neutral, hence not of the form $\lambda x.v$ (it cannot be $rep(U_1 \rightarrow U_2)$ as we already assumed $U_1 \rightarrow U_2 \notin Iso(\mathbf{T})$).

□

A.1 Reducibility with parameters

Let T be a type, and let $\overrightarrow{X}$ be a set of type variables containing at least all the free type variables of T. For $\overrightarrow{U}$ a sequence of types of the same length, let $T[\overrightarrow{U}/\overrightarrow{X}]$ be the type obtained by simultaneous substitution of the X's with the U's, and let $\overrightarrow{R}$ a sequence of reducibility candidates of corresponding types.

Definition A.9
The set $RED_T[\overrightarrow{R}/\overrightarrow{X}]$ of reducible terms of type $T[\overrightarrow{U}/\overrightarrow{X}]$ is defined by reducibility with parameters induction on the type T as follows.

- if T is atomic, $RED_T[\overrightarrow{R}/\overrightarrow{X}]$ is the set of strongly normalizable terms of type $T[\overrightarrow{U}/\overrightarrow{X}] = T$
- if T is X_i, $RED_T[\overrightarrow{R}/\overrightarrow{X}]$ is R_i
- if T is $U \times V$, then $RED_T[\overrightarrow{R}/\overrightarrow{X}]$ is $RED_U[\overrightarrow{R}/\overrightarrow{X}] \times RED_V[\overrightarrow{R}/\overrightarrow{X}]$
- if T is $U \to V$, then $RED_T[\overrightarrow{R}/\overrightarrow{X}]$ is $RED_U[\overrightarrow{R}/\overrightarrow{X}] \to RED_V[\overrightarrow{R}/\overrightarrow{X}]$
- if T is $\forall Y.W$, then $RED_T[\overrightarrow{R}/\overrightarrow{X}]$ is the set of terms t of type $[\overrightarrow{U}/\overrightarrow{X}]$ such that, for every type V and reducibility candidate S of this type, t[V] $\in$ $RED_W[\overrightarrow{R}/\overrightarrow{U}, S/Y]$

Lemma A.10
$rep(U)$ is normal for all $U \in Iso(\mathbf{T})$.

Proof
By a straightforward induction on the structure of the term. □

Theorem A.11
$RED_T[\overrightarrow{R}/\overrightarrow{X}]$ is a reducibility candidate of type $T[\overrightarrow{U}/\overrightarrow{X}]$

Proof
We proceed by structural induction on the type T.

Since we consider only terms in *gentop* normal form, *there is no term of type U besides $rep(U)$ if $U \in Iso(\mathbf{T})$*. Moreover, due to the previous lemma and the definition of reducibility, $rep(U)$ trivially satisfies (CR1), (CR2) and (CR3), so we will not consider explicitly the case of types in $Iso(\mathbf{T})$ in the induction.

Atomic types

If T is atomic, then $RED_T[\overrightarrow{R}/\overrightarrow{X}]$ is the set of strongly normalizing terms of type T, and we already proved it to be a reducibility candidate (Proposition A.5).

Type variables

If T is X_i, then $RED_T[\overrightarrow{R}/\overrightarrow{X}]$ is R_i, that is a reducibility candidate by definition.

https://doi.org/10.1017/S0956796800001696 Published online by Cambridge University Press

Product types

Let T be $U_1 \times U_2$. Then $RED_T[\vec{R}/\vec{X}] = RED_{U_1}[\vec{R}/\vec{X}] \times RED_{U_1}[\vec{R}/\vec{X}]$ by definition. We can apply the induction hypothesis for $RED_{U_1}[\vec{R}/\vec{X}]$ and $RED_{U_2}[\vec{R}/\vec{X}]$, so that the result then follows by Theorem A.8.

Arrow types

Let T be $U_1 \to U_2$. Then $RED_T[\vec{R}/\vec{X}] = RED_{U_1}[\vec{R}/\vec{X}] \to RED_{U_1}[\vec{R}/\vec{X}]$ by definition. We can apply the induction hypothesis for $RED_{U_1}[\vec{R}/\vec{X}]$ and $RED_{U_2}[\vec{R}/\vec{X}]$, so that the result then follows by Theorem A.8.

Universal types

Let $T = \forall Y.W$. We can assume that $W \notin Iso(\mathbf{T})$ as otherwise $\forall Y.W \in Iso(\mathbf{T})$.

- (CR1) if $t \in RED_{\forall Y.W}[\vec{R}/\vec{X}]$, then let V be an arbitrary type and S be an arbitrary reducibility candidate of this type (for example, the strongly normalizable terms of type V). Then t[V] $\in RED_W[\vec{R}/\vec{X}, S/Y]$, and so, by induction hypothesis, we know that t[V] is strongly normalizable. A fortiori t is strongly normalisable.
- (CR2) if $t \stackrel{\beta\eta\pi *}{\longrightarrow} t'$, then for all types V and reducibility candidate S of this type, we have that t[V] $\in RED_W[\vec{R}/\vec{X}, S/Y]$ and (t[V]) $\stackrel{\beta\eta\pi *}{\longrightarrow}$ (t'[V]), hence t'[V] $\in RED_W[\vec{R}/\vec{X}, S/Y]$ by induction hypothesis on W. So, by definition, $t' \in RED_{\forall Y.W}[\vec{R}/\vec{X}]$.
- (CR3) t is neutral and all t' one step from t are in $RED_T[\vec{R}/\vec{X}]$. Take V and S: if we apply a conversion inside $t[V]$, the result is $t'[V]$ since t is neutral (and, again, not $rep(\forall Y.W)$, as $t \stackrel{\beta\eta\pi *}{\longrightarrow}$ t'). Now, $t'[V]$ is in $RED_W[\vec{R}/\vec{X}, S/Y]$ as t' is in $RED_T[\vec{R}/\vec{X}]$. By induction hypothesis, we get $t'[V] \in RED_W[\vec{R}/\vec{X}, S/Y]$, so $t \in RED_T[\vec{R}/\vec{X}]$.

□

Reducibility theorem

We shall need some lemmas to deduce reducibility of a term from reducibility of its subterms.

Lemma A.12
(Pairing) Let $u_1 \in RED_{U_1}[\vec{R}/\vec{X}]$ and $u_2 \in RED_{U_2}[\vec{R}/\vec{X}]$.
Then $\langle u_1, u_2 \rangle \in RED_{U_1 \times U_2}[\vec{R}/\vec{X}]$.

Proof
We can assume that $U_1 \notin Iso(\mathbf{T})$ and/or $U_2 \notin Iso(\mathbf{T})$, as otherwise $\langle u_1, u_2 \rangle = rep(U_1 \times U_2)$ and then $RED_{U_1 \times U_2}[\vec{R}/\vec{X}]$ is $\{rep(U_1 \times U_2)\}$.

We can argue by induction on $\nu(u_1) + \nu(u_2)$, by CR1, to show that, for $i = 1$ and/or $i = 2$, $p_i\langle u_1, u_2\rangle \in RED_{U_i}[\overrightarrow{R}/\overrightarrow{X}]$.
Let $i = 1$ for simplicity. The term $p_1\langle u_1, u_2\rangle$ converts to:

- u_1, which is in $RED_{U_1}[\overrightarrow{R}/\overrightarrow{X}]$ by hypothesis.
- $p_1\langle u', u_2\rangle$ with u' one step from u_1.
  Then u' is in $RED_{U_1}[\overrightarrow{R}/\overrightarrow{X}]$ by CR2 and ν(u') < $\nu(u_1)$, so $p_1\langle u', u_2\rangle \in RED_{U_1}[\overrightarrow{R}/\overrightarrow{X}]$ by induction hypothesis.
- $p_1\langle u_1, v'\rangle$ with v' one step from u_2. We get $p_1\langle u_1, v'\rangle \in RED_{U_1}[\overrightarrow{R}/\overrightarrow{X}]$ as above.
- p_1w if u_1 is p_1w and u_2 is p_2w.
  But p_1w $= u_1$ is in $RED_{U_1}[\overrightarrow{R}/\overrightarrow{X}]$ by hypothesis.
- p_1w if u_1 is p_1w and u_2 is $rep(U_2)$.
  By definition of parametric reducibility for product types when one of the factor types is in $Iso(\mathbf{T})$, we get that $u_1 \in RED_{U_1}[\overrightarrow{R}/\overrightarrow{X}]$ as p_1w $= u_1$ is in $RED_{U_1}[\overrightarrow{R}/\overrightarrow{X}]$ by hypothesis.

In every case, the neutral terms $p_i\langle u_1, u_2\rangle$ convert to terms in $RED_{U_i}[\overrightarrow{R}/\overrightarrow{X}]$ only, for $i = 1$ and/or $i = 2$, so they are in $RED_{U_i}[\overrightarrow{R}/\overrightarrow{X}]$ by CR3. Hence $\langle u_1, u_2\rangle$ is in $RED_{U_1 \times U_2}[\overrightarrow{R}/\overrightarrow{X}]$. $\square$

Lemma A.13
(Abstraction) Let $x{:}U$ and v:V. If for all $u \in RED_U[\overrightarrow{R}/\overrightarrow{X}]$ we have that $v[u/x] \in RED_V[\overrightarrow{R}/\overrightarrow{X}]$, then $\lambda x.v \in RED_{U\to V}[\overrightarrow{R}/\overrightarrow{X}]$.

Proof
We can assume that $V \notin Iso(\mathbf{T})$ as otherwise v is $rep(V)$, and $\lambda x.v$ is $rep(U \to V)$ as $U \to V \in Iso(\mathbf{T})$, and it is reducible by definition.

To show that $\lambda x.v \in RED_{U\to V}[\overrightarrow{R}/\overrightarrow{X}]$, we need to show that $(\lambda x.v)u \in RED_V[\overrightarrow{R}/\overrightarrow{X}]$ for all $u \in RED_U[\overrightarrow{R}/\overrightarrow{X}]$.
There are two cases: either $U \in Iso(\mathbf{T})$ or not.
In the first case, $v[u/x] = v$ as it is in *gentop* normal form, hence there is no free occurrence of x in v, and the only term u of type U is $rep(U)$. Since $t = (\lambda x.v)u$ is neutral, it suffices to show that for every term t' one-step from it $t' \in RED_V[\overrightarrow{R}/\overrightarrow{X}]$. Since $v = v[rep(U)/x] \in RED_V[\overrightarrow{R}/\overrightarrow{X}]$ by hypothesis, hence strongly normalizing, we can argue by induction on ν(v). The one-step reducts of $(\lambda x.v)$u are:

- $v[u/x]$ which is in $RED_V[\overrightarrow{R}/\overrightarrow{X}]$ by hypothesis
- $(\lambda x.v')u$ with v' one step from v. Then $v'[u/x]$ is in $RED_V[\overrightarrow{R}/\overrightarrow{X}]$ by CR2 as it is one step from $v[u/x]$ and we are done by induction hypothesis as ν(v') < ν(v)
- $(v'u)$ via η_{top} if $v = v'rep(U)$.
  Now, $u = rep(U)$ so $(v'u) = v'rep(U) = v = v[u/x]$ which is in $RED_V[\overrightarrow{R}/\overrightarrow{X}]$ by hypothesis.

In the second case, $x{:}U$ is in $RED_U[\overrightarrow{R}/\overrightarrow{X}]$ (Remark A.4). So $v = v[x/x]$ is in $RED_V[\overrightarrow{R}/\overrightarrow{X}]$ and hence strongly normalizable by CR2, and we can argue by

https://doi.org/10.1017/S0956796800001696 Published online by Cambridge University Press

induction on $\nu(\mathrm{u}) + \nu(\mathrm{v})$ to show that all terms one step from $(\lambda x.vu)$ are reducible. The one-step reducts of$(\lambda x.v)u$ are:

- $v[u/x]$ that is in $RED_V[\vec{R}/\vec{X}]$ by hypothesis.
- $(\lambda x.v')u$ with v' one step from v. Since $v'[u/x]$ is one step from $v[u/x]$[9], then it is in $RED_V[\vec{R}/\vec{X}]$ by CR2. Furthermore, $\nu(\mathrm{v'}) < \nu(\mathrm{v})$, so by induction hypothesis we get $(\lambda x.v'u) \in RED_V[\vec{R}/\vec{X}]$.
- $(\lambda x.v)u'$ with u' one step from u. Then $u' \in RED_U[\vec{R}/\vec{X}]$ by CR2, $\nu(\mathrm{u'}) < \nu(\mathrm{u})$ and $v[u'/x] \in RED_V[\vec{R}/\vec{X}]$ by repeated applications of CR2, as it is some step from $\mathrm{v}[u/x]$. So we can apply again the induction hypothesis.
- $(v'u)$ via η if $\lambda x.v$ is $\lambda x.v'x$ and $x \notin \mathrm{FV(v')}$.
  It is in $RED_V[\vec{R}/\vec{X}]$ as $v[u/x] = (v'u)$ is in $RED_V[\vec{R}/\vec{X}]$ by hypothesis.

Since $(\lambda x.v)u$ is neutral and it converts to reducible terms only, it is reducible. Hence $\lambda x.v$ is reducible. □

Remark A.14
Working only with terms in *gentop* normal form allows us to rule out all the other reductions that are possible when considering all the terms of the calculus. This restriction is essential since otherwise we ought now to face, in Lemma A.12, reductions like $p_1\langle rep(U_1), p_2w\rangle \longrightarrow p_1\mathrm{w}$, that we cannot handle, for nothing in our induction hypothesis allows us to conclude that $p_1\mathrm{w}$ is reducible. (We already pointed out the difficulty in Section 2.3.) This reduction[10] is now ruled out as $p_1\langle rep(U_1), p_2w\rangle$ is not a *gentop* normal form (its normal form being $rep(U_1)$). Similarly, in Lemma A.13, the restriction to terms in *gentop* normal form allows us to rule out (in the case $U \in Iso(\mathbf{T})$) all the other reductions otherwise possible in the full calculus. As pointed out in the introduction (Section 2.3), we do not know how to handle the general reduction $(\lambda x.(v'rep(U)))u \longrightarrow (v'u)$ via η_{top}: if u is not $rep(U)$, then we have nothing in our induction hypothesis to tell us that $(v'u)$ is reducible. But here u *must* be in *gentop* normal form, that is to say, $u = rep(U)$, and the η_{top} reduction can be handled as above.

Lemma A.15
(Universal abstraction) If for every type V and candidate S of type V, $v[V/Y] \in RED_W[\vec{R}/\vec{X}, S/Y]$, then $\Lambda Y.v \in RED_{\forall Y.W}[\vec{R}/\vec{X}]$.

Proof
We need to show that $(\Lambda Y.v)[V] \in RED_W[\vec{R}/\vec{X}, S/Y]$ for every type V and candidate S of type V. We argue by induction on $\nu(\mathrm{v})$, using the fact that $(\Lambda Y.\mathrm{v})[V]$ is neutral. Converting a redex of $(\Lambda Y.\mathrm{v})[V]$ can yield:

- $(\Lambda Y.\mathrm{v'})[V]$ with v' one step from v; now, by induction hypothesis on $\nu(\mathrm{v})$, we know that $(\Lambda Y.v')[V] \in RED_W[\vec{R}/\vec{X}, S/Y]$.

The result follows by CR3. □

[9] Can be shown by an easy induction on v.
[10] And its symmetric $p_2\langle p_1w, rep(U_2)\rangle \longrightarrow p_2\mathrm{w}$.

Lemma A.16
$RED_{T[V/Y]}[\vec{R}/\vec{X}] = RED_T[\vec{R}/\vec{X}, RED_V[\vec{R}/\vec{X}]/Y]$

Proof
By induction on T. □

Lemma A.17
(Universal application) If $t \in RED_{\forall Y.W}[\vec{R}/\vec{X}]$, then $t[V] \in RED_{W[V/Y]}[\vec{R}/\vec{X}]$ for every type V.

Proof
By hypothesis, $t[V] \in RED_W[\vec{R}/\vec{X}, S/Y]$ for every candidate S. Taking $S = RED_V[\vec{R}/\vec{X}]$, the result follows by Lemma A.16. □

The theorem

As in Girard *et al.* (1990), we say here that a term t of type T is *reducible* if it is in $RED_T[\vec{SN}/\vec{X}]$, where $\vec{X}$ are the free type variables of T and SN_i is the set of strongly normalizable terms of type X_i. In the proof of the theorem, there is the need of a stronger induction hypothesis, from which the strong normalization follows by putting $u_i = x_i$ and $R_i = SN_i$.

Proposition A.18
Let t:T be any term of $\lambda^2\beta\eta\pi*$ (in *gentop* normal form), whose free variables are among $x_1 : U_1, \ldots, x_n : U_n$, and all the free variables of T, $U_1, \cdots U_n$ are among $X_1, \cdots X_m$. If $R_1, \ldots R_m$ are reducibility candidates of types $V_1, \cdots V_m$, and $u_1, \cdots$, u_m are terms of types $U_1[\vec{V}/\vec{X}], \ldots U_m[\vec{V}/\vec{X}]$ which are in $RED_{U_1}[\vec{R}/\vec{X}], \ldots$, $RED_{U_n}[\vec{R}/\vec{X}]$, then $t[\vec{V}/\vec{X}][\vec{u}/\vec{x}] \in RED_T[\vec{R}/\vec{X}]$.

Proof
By induction on t. Notice that there are no variables of type U if $U \in Iso(\mathbf{T})$.

- $t = *$: t is in the only reducibility candidate $\{*\}$ of type $\mathbf{T}$.
- $t = x_i$: in this case the statement of the theorem becomes a tautology.
- $t = p_i$u : then $u : U_1 \times U_2$ and $U_i \notin Iso(\mathbf{T})$ as we consider only terms in *gentop* normal form. By induction hypothesis, $u[\vec{V}/\vec{X}][\vec{u}/\vec{x}] \in RED_{U_1 \times U_2}[\vec{R}/\vec{X}]$. Hence $(p_i u)[\vec{V}/\vec{X}][\vec{u}/\vec{x}] = p_i u[\vec{V}/\vec{X}][\vec{u}/\vec{x}] \in RED_{U_i}[\vec{R}/\vec{X}]$ by definition of reducibility for product types.
- $t = \langle u, v\rangle$: $u[\vec{V}/\vec{X}][\vec{u}/\vec{x}] \in RED_{U_1}[\vec{R}/\vec{X}]$ and $v[\vec{V}/\vec{X}][\vec{u}/\vec{x}] \in RED_{U_2}[\vec{R}/\vec{x}]$ by the induction hypothesis, so Lemma A.12 gives $\langle u[\vec{V}/\vec{X}][\vec{u}/\vec{x}], v[\vec{V}/\vec{X}][\vec{u}/\vec{x}]\rangle \in RED_{U_1 \times U_2}[\vec{R}/\vec{X}]$. Now, $\langle u, v\rangle$ $[\vec{V}/\vec{X}][\vec{u}/\vec{x}]$ is $\langle u[\vec{V}/\vec{X}][\vec{u}/\vec{x}], v[\vec{V}/\vec{X}][\vec{u}/\vec{x}]\rangle$, and hence $\langle u, v\rangle[\vec{V}/\vec{X}][\vec{u}/\vec{x}] \in RED_{U_1 \times U_2}[\vec{R}/\vec{X}]$.
- $t = \lambda z.v$: by induction hypothesis, we know that $v[\vec{V}/\vec{X}][\vec{u}/\vec{x}][u/z] \in RED_V[\vec{R}/\vec{X}]$ for all $u \in RED_U[\vec{R}/\vec{X}]$. Then Lemma A.13 gives $\lambda z.v[\vec{V}/\vec{X}][\vec{u}/\vec{x}] \in RED_{U \to V}[\vec{R}/\vec{X}]$. But $(\lambda z.v)[\vec{V}/\vec{X}][\vec{u}/\vec{x}]$ is $\lambda z.v[\vec{V}/\vec{X}][\vec{u}/\vec{x}]$ by definition, and the result follows.

- $t = vu$: then $v[\vec{V}/\vec{X}][\vec{u}/\vec{x}] \in RED_{U\to V}[\vec{R}/\vec{X}]$, so $u[\vec{V}/\vec{X}][\vec{u}/\vec{x}] \in RED_U[\vec{R}/\vec{X}]$ by induction hypothesis. Hence we know that $(v[\vec{V}/\vec{X}][\vec{u}/\vec{x}]\ u[\vec{V}/\vec{X}][\vec{u}/\vec{x}]) \in V$, as it is $(vu)[\vec{V}/\vec{X}][\vec{u}/\vec{x}]$ by definition.
- $t = \Lambda Y.v$: then we know by induction hypothesis that for every type V and reducibility candidate S we have $v[V/Y][\vec{V}/\vec{X}][\vec{u}/\vec{x}] \in RED_W[\vec{R}/\vec{X}, S/Y]$. Then, applying Lemma A.15, we get that $(\Lambda Y.v)[\vec{V}/\vec{X}][\vec{u}/\vec{x}] \in \forall Y.W$.
- $t = t[V]$: then we know by induction hypothesis that $t[\vec{V}/\vec{X}][\vec{u}/\vec{x}] \in RED_{\forall Y.W}[\vec{R}/\vec{X}]$ and, by Lemma A.17, for every type V $t[V][\vec{V}/\vec{X}][\vec{u}/\vec{x}] \in RED_{W[V/Y]}[\vec{R}/\vec{X}]$.

□

Theorem A.19
$\xrightarrow{\beta\eta^2\pi*}$ is strongly normalizing over the set of *gentop* normal forms.

Proof
Let t be any term in *gentop* normal form. All its free variables are in any reducibility candidate by CR3, so that $t = t[\vec{SN}/\vec{X}][\vec{x}/\vec{x}]$ is reducible by the previous lemma. By CR1 it is strongly normalizing. That is, $\xrightarrow{\beta\eta^2\pi*}$ is strongly normalizing over *gentop* normal forms. □

B Normalization without η_{top} and SP_{top}

The proof of strong normalization is essentially the same as the one given above for the full system without β^2 over the subset of terms in *gentop* normal form.

The main difference, besides the fact that we add β^2 and *gentop* and exclude η_{top} and SP_{top}, is that now we work on the full set of terms, so that there are plenty of terms $t{:}U$, besides $rep(U)$, when $U \in Iso(\mathbf{T})$. We keep essentially the same notion of neutral term (A.1), but it is to be noted that only $rep(U)$ is neutral, not every term of type $U \in Iso(\mathbf{T})$.

Definition B.1 (neutral terms)
A term $t : U$ is neutral iff at least one of the following conditions is satisfied:

- if $U \notin Iso(\mathbf{T})$ and t is not an abstraction, a type abstraction or a pair,
- if $U \in Iso(\mathbf{T})$ and t is $rep(U)$.

Since we drop η_{top} and SP_{top}, there is no need to give a special status to the types $U \in Iso(\mathbf{T})$ (besides the fact that $rep(U)$ is neutral), and we resort to the usual definition of product and function space of reducibility candidates, which allows us to deal with all the terms of type $U \in Iso(\mathbf{T})$.

Definition B.2 (product and arrow of reducibility candidates)
If R and S are reducibility candidates of types U and V, we define:

- $t \in R \to S \iff$ for all $u \in R$, $tu \in S$
- $t \in R \times S \iff p_1 t \in U$ and $p_2 t \in V$

With this new definition, the proofs of the previous appendix go through almost unchanged, with the only care to keep in mind that now $rep(U)$ is no longer the only term of type $U \in Iso(\mathbf{T})$, and that types in $Iso(\mathbf{T})$ have no longer a special status. This means that wherever there is a distinction between types that are in $Iso(\mathbf{T})$ and types that are not, one follows the proof given for types that are not in $Iso(\mathbf{T})$. The new cases arising from *gentop* reductions are easily dealt with, as $rep(U)$ is still in any reducibility candidate by CR3.

For completeness, we detail here all the changes that are needed.

- Remark A.4 now extends to *all* variables and also the variables of type $U \in Iso(\mathbf{T})$. It is just the matter of noticing that a variable $x{:}U \in Iso(\mathbf{T})$ is neutral and reduces only to $rep(U)$, that is, in any reducibility candidate by CR3, and the result follows by CR3.
- In Theorem A.8, we can no longer factor out the types in $Iso(\mathbf{T})$, that must be treated exactly as the other types:

  Product Types (CR3)

  – t can be $rep(U_1 \times U_2)$. In that case the only possible reduction for p_it (that is not in *gentop* normal form) is to $rep(U_i)$, that belongs to all reducibility candidate (Remark A.4), hence in $RED_{U_i}[\vec{R}/\vec{X}]$ that is a reducibility candidate by induction hypothesis on U_i. So $p_i t \in RED_{U_i}[\vec{R}/\vec{X}]$ by CR3 on U_i and we get $t \in RED_{U_1 \times U_2}[\vec{R}/\vec{X}]$ by definition.

  – t can be a neutral term different from $rep(U_1 \times U_2)$. Then the only possible reduction for p_it (that is not in *gentop* normal form) is to $rep(U_i)$, and we conclude as above.

  Arrow Types (CR3)

  – t (or t') can be $rep(U_1 \to U_2)$. Then (tu) (or $(t'u)$) can only reduce to $rep(U_2)$ that is in any reducibility candidate (Remark A.4), hence in $RED_{U_2}[\vec{R}/\vec{X}]$ that is a reducibility candidate by induction hypothesis on U_2. So (tu)(or $(t'u)$) $\in RED_{U_2}[\vec{R}/\vec{X}]$ for all $u \in RED_{U_1}[\vec{R}/\vec{X}]$, and we get $t \in RED_{U_1 \to U_2}[\vec{R}/\vec{X}]$ by definition.

  – t can be a neutral term different from $rep(U_1 \to U_2)$. Then the only possible reduction for (tu) (or $(t'u)$) is to $rep(U_2)$, and we conclude as above.

- In Theorem A.11, we can no longer factor out the types in $Iso(\mathbf{T})$ that must be treated exactly as the other types.

  Universal Types (CR3)

  – t (or t') can be $rep(\forall Y.W)$. Then $t[V]$ can only reduce to $rep(W)$, that is in any reducibility candidate (Remark A.4), hence in $RED_W[\vec{R}/\vec{X}]$ that belongs to all reducibility candidate by induction hypothesis on W. Again we get t(or t') $\in RED_{\forall Y.W}[\vec{R}/\vec{X}]$ by definition.

  – t (or t') can be a neutral term different from $rep(\forall Y.W)$. Then $t[V]$ can only reduce to $rep(W)$, and we conclude as above.

- In Lemmas A.12 and A.13 we can no longer factor out the case of types $U \in Iso(\mathbf{T})$, which must be treated uniformly as the other types. Since the rules SP_{top} and η_{top} are not present, only the first four cases considered in Lemma A.12 can occur, and the proof goes through unchanged for them, while for Lemma A.13 we follow the proof given for $V \notin Iso(\mathbf{T})$.
  There is now the further possibility of a *gentop* reduction, that in both cases is dealt with in the usual way by remembering that *any* reducibility candidate of type $U \in Iso(\mathbf{T})$ contains $rep(U)$.
- In Lemma A.15 we have now two additional cases:
  — $(\Lambda Y.v)[V]$ reduces to the term $rep(\mathrm{W}[V/Y])$, that must belong to $RED_{W[V/Y]}[\vec{R}/\vec{X}]$ since this latter is a reducibility candidate.
  — $(\Lambda Y.v)[V]$ reduces to $\mathrm{v}[V/Y]$. But we know by hypothesis that $v[V/Y] \in RED_{W[V/Y]}[\vec{R}/\vec{X}, S/Y]$
- In the proof of the Proposition A.18, it suffices to apply to the types $V \in Iso(\mathbf{T})$ the same arguments used for types $U \notin Iso(\mathbf{T})$, as now there is no longer any difference in the definition of the function space and product of reducibility candidates.

Using again the fact that $\mathrm{t}=\mathrm{t}[\vec{SN}/\vec{X}][\vec{x}/\vec{x}]$, we similarly get our final result.

Theorem B.3
$\xrightarrow{\beta^2\eta^2\pi*}$ without η_{top} and SP_{top} is strongly normalizing.

References

Akama, Y. (1993) On Mints' reductions for ccc-Calculus. *Typed lambda calculus and applications. LNCS 664*, pp. 1–12. Springer-Verlag.

Barendregt, H. (1984) *The lambda calculus; its syntax and semantics (revised edition).* North Holland.

Breazu-Tannen, V. (1988) Combining algebra and higher order types. In: *Proceedings of the symposium on logic in computer science (LICS)*, pp. 82–90.

Breazu-Tannen, V. and Gallier, J. (1994) Polymorphic rewiting preserves algebraic confluence. *Information and Computation.* To appear.

Bruce, K., Di Cosmo, R. and Longo, G. (1992) Provable isomorphisms of types. *Mathematical Structures in Computer Science*, **2**(2), 231–247.

Cubric, D. (1992) *On free CCC.* Distributed on the types mailing list.

Curien, P.-L. and Ghelli, G. (1990) *Confluence and decidability of $\beta\eta top_{\leq}$ reduction on $F_{\leq}$.* *Information and Computation.* To appear.

de Vrijer, R. C. (1987) *Surjective pairing and strong normalization: two themes in λ-calculus.* Ph.D. thesis, Universiteit van Amsterdam.

Dezani-Ciancaglini, M. (1976) Characterization of normal forms possessing an inverse in the $\lambda\beta\eta$ calculus. *Theoretical Computer Science*, **2**, 323–337.

Di Cosmo, R. (1994) Second order isomorphic types. A proof theoretic study on second order λ-calculus with surjective pairing and terminal object. *Information and Computation.* To appear.

Di Cosmo, R. and Kesner, D. (1994a) Combining first order algebraic rewriting systems, recursion and extensional lambda calculi. In: Abiteboul, S. and Shamir, E. (eds), *International Conference on Automata, Languages and Programming (ICALP)*, pp. 462–472. *Lecture Notes in Computer Science, 820.* Springer-Verlag.

Di Cosmo, Ro. and Kesner, D. (1994b) Simulating expansions without expansions. *Mathematical Structures in Computer Science*, **4**, 1–48.

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

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

Hardin, T. (1989) Confluence results for the pure strong categorical logic C.C.L.; λ-calculi as subsystems of C.C.L. *Theoretical Computer Science*, **65**(2), 291–342.

Jay, C. B. and Ghani, N. (1992) *The Virtues of Eta-expansion.* Tech. rept. ECS-LFCS-92-243. LFCS. University of Edimburgh.

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

Lambek, J. and Scott, P. J. (1986) *An Introduction to Higher Order Categorical Logic.* Cambridge University Press.

Mints, G. *A simple proof of the coherence theorem for cartesian closed categories. Bibliopolis.* To appear.

Nipkow, T. (1990) A critical pair lemma for higher-order rewrite systems and its application to λ^*. In: *First Annual Workshop on Logical Frameworks.*

Obtulowicz, A. (1987) Algebra of constructions I. The Word Problem for Partial Algebras. *Information and Computation*, **73**(2), 129–173.

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

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

Tait, W. W. (1967) Intensional interpretation of functionals of finite type I. *Journal of Symbolic Logic*, **32**.

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