# $\beta\eta$ -Equality for Coproducts

# NEIL GHANI

LIENS-DMI, Ecole Normale Superieure, 45 Rue D'Ulm, 75230 Paris Cedex 05, France. email: ghani@ens.fr

Received 3 May 0

The use of expansionary  $\eta$ -rewrite rules in various typed  $\lambda$ -calculi has become increasingly common in recent years as their advantages over contractive  $\eta$ -rewrite rules have become apparent. Not only does one obtain simultaneously a decision procedure for  $\beta\eta$ -equality and a rational reconstruction of the long  $\beta\eta$ -normal forms, but expansions retain key properties such as strong normalisation and confluence when combined with algebraic rewrite systems, are supported by a categorical theory of reduction and generalise more easily to other type constructors.

This paper considers a type constructor for which a decision procedure for  $\beta\eta$ -equality has been sought for a long time, namely the coproduct. Categorical models of reduction are used to derive a new  $\eta$ -rewrite rule for the coproduct which turns out to be substantially more complex than that for the exponent or product. Not only is there a facility for expanding terms of sum type analogous to that for the product and exponential, but also the ability to permute the order in which different subterms of sum type occur.

These different aspects of  $\eta$ -conversion for the sum type are reflected in our analysis. The rewrite relation is decomposed into two parts, a strongly normalising and confluent fragment resembling that found in the calculus without coproducts and a relation which generalises the "commuting conversions" appearing in the literature. This second fragment is proved decidable by constructing for each term its (finite) set of *quasi-normal reducts*. Finally decidability, confluence and quasi-normal forms for the full relation are derived by embedding the whole relation into this generalised commuting conversion relation.

### 1. Introduction

Extensional equality for terms of the simply typed  $\lambda$ -calculus requires  $\eta$ -conversion, whose interpretation as a rewrite rule has traditionally been as a contraction  $\lambda x.fx \Rightarrow f$  with the side condition  $x \not\in FV(f)$ . When combined with the usual  $\beta$ -reduction, the resulting rewrite relation is strongly normalising and confluent, and thus reduction to normal form provides a decision procedure for the associated equational theory.

However,  $\eta$ -contractions behave badly when combined with other rewrite rules and the key property of confluence is often lost. For example, if the calculus is extended by a unit type 1 with associated rewrite rule  $t \Rightarrow *$  (providing t has type 1), then the divergence

$$\lambda x : 1. * \Leftarrow \lambda x : 1. f x \Rightarrow f \tag{1}$$

cannot be completed.

Another area where  $\eta$ -contractions cannot be used is in the combination of type theories and algebraic rewrite systems. Properties of such rewrite systems such as confluence and strong normalisation, preserved when combined with  $\beta$ -reduction (V. Breazu-Tannen 1988; V. Breazu-Tannen and J. Gallier 1994), are typically lost in the presence of a contractive  $\eta$ -rewrite rule. For example, if we regard 1 as a base type with constants  $f: 1 \rightarrow 1$ and \*: 1 and with rewrite rule  $fx \Rightarrow *$ , then  $\Rightarrow$  is confluent while the divergence above shows that the combination of  $\Rightarrow$  with the contractive  $\eta$ -rewrite rule is not confluent.

These problems led several authors (Y. Akama 1993; R. Di Cosmo and D. Kesner 1994; C. B. Jay and N. Ghani 1995) to accept the old proposal (G. Huet 1976; G. E. Mints 1979; D. Prawitz 1971) that  $\eta$ -conversion be interpreted as an expansion  $f \Rightarrow \lambda x.fx$  and the resulting rewrite relation has been shown confluent. In these works infinite reduction sequences such as

$$f \Rightarrow \lambda x \cdot f x \Rightarrow \lambda x \cdot (\lambda y \cdot f y) x \Rightarrow \dots$$

are avoided by imposing syntactic restrictions to limit the possibilities for expansion; namely  $\lambda$ -abstractions cannot be expanded, nor can terms which are applied. This restricted expansion relation is strongly normalising, confluent and generates the same equational theory as the unrestricted expansionary rewrite relation. Thus  $\beta\eta$ -equality can be decided by reduction to normal form in this restricted fragment and, in addition, the normal forms of this restricted rewrite relation are exactly Huet's long  $\beta\eta$ -normal forms (G. Huet 1976; D. Prawitz 1971). In addition,  $\eta$ -expansions generalise well to the powerfull members of the  $\lambda$ -cube (N. Ghani 1995a; N. Ghani 1996) and, most pleasingly of all, these properties tend to be maintained if one adds algebraic rewrite rules (R. Di Cosmo and D. Kesner 1994).

In addition to these practical arguments, the category-theoretic analysis of reduction (N. Ghani 1995b; C. B. Jay 1992; D. E. Rydeheard and J. G. Stell 1987; R. A. G. Seely 1987) provides another argument in favour of interpreting  $\eta$  as an expansion. In this analysis, the introduction and elimination rules of a type constructor form a pair of locally adjoint functors (J. Gray 1974; C. B. Jay 1988) whose local unit and counit are respectively an expansionary (not *contractive*)  $\eta$ -rewrite rule and contractive  $\beta$ -rewrite rule. The associated local triangle laws assert the existence of looping reductions — for the exponential the triangle laws are

Thus even the restrictions on  $\eta$ -expansion required to obtain strong normalisation have a categorical formulation, preventing exactly those expansions occurring in the triangle laws 2.

This paper considers a type constructor for which a decision procedure for  $\beta\eta$ -equality has been sought for a long time, namely the coproduct or sum type. The categorical approach to rewriting outlined above is used to derive a new  $\eta$ -rewrite rule for the coproduct which turns out to be substantially more complex than that for the exponent or product. Not only is there a facility for expanding terms of sum type analogous to that for the product and exponential, but also the ability to permute the order in which different subterms of sum type occur.

These different aspects of  $\eta$ -expansion for the sum type are reflected in our analysis. After defining the calculus, an expansionary  $\eta$ - and a contractive  $\beta$ -rewrite rule is derived for each type constructor by interpreting the associated introduction and elimination rules as forming an adjoint pair. This rewrite relation is then decomposed into two fragments, the first of which contains  $\beta$ -redexes, commuting conversions and limited possibilities for  $\eta$ -expansion and is proven strongly normalising and confluent. The normal forms of this fragment satisfy similar structural criteria to the long  $\beta\eta$ -normal forms of the simply typed  $\lambda$ -calculus, and so may be thought of as their generalisation to this calculus.

The second part of the decomposition is called the *conversion relation* and permutes the order in which subterms of sum type may be eliminated — examples of which are the 'commuting conversions' appearing in the literature (D. Prawitz 1971; J. Y. Girard *et al.* 1989). Each term has a finite set of possible permutations, and so in general unique normal forms do not exist for the conversion relation. Instead, each term has a (finite) set of *quasinormal reducts* and terms equivalent in the equational theory generated by the conversion relation have the same set of quasi-normal reducts. Confluence and decidability of the conversion relation in the conversion relation, confluence and decidability of the full rewrite relation is proved.

Historically the use of expansionary  $\eta$ -rewrite rules for products and exponentials can be traced back to (G. E. Mints 1979), although the proof that they form a strongly normalising relation had to wait a decade for the papers mentioned above. The last year has seen the successful application of  $\eta$ -expansions to more powerful theories in the  $\lambda$ cube (N. Ghani 1995a; N. Ghani 1996), and currently research focuses on combining these powerful type theories with algebraic rewrite systems. In (N. Ghani 1995b), the methods presented in this paper are used to define, and prove decidable, a sound and complete equational theory for the  $(I, \otimes, \rightarrow)$ -fragment of intuitionistic linear logic.

Several authors have attempted to apply  $\eta$ -expansions to the problem of  $\beta\eta$ -equality for coproducts. A partial solution was provided by (D. Dougherty 1993) but in this approach confluence can only be proved for terms of ground type. At the time of writing, the research presented here remains the only proof of the decidability of the theory of coproducts, although one other interesting result is (D. Dougherty and R. Subrahmanyam 1995) which extends (H. Friedman 1975) in providing a proof system for deriving a set of equations which is sound and complete for all "set-theoretic" models of a  $\lambda$ -calculus with exponentials and coproducts. This theory has been proved decidable by proving it is equivalent to the one presented here (D. Dougherty and R. Subrahmanyam 1995).

The rest of this paper is organised as follows. Section 2 contains notation required later, section 3 a definition of the term calculus and section 4 uses categorical methods to derive a rewrite relation which generates a sound and complete equality. Section 5 defines the conversion relation, while sections 6 and 7 prove the conversion relation decidable.

Section 8 defines, and proves strongly normalising, the extension of  $\beta$ -reduction while section 9 combines these results to prove the full relation is decidable. Finally we make some concluding remarks in section 10.

# 2. Notation

While basic knowledge of term rewriting is assumed (N. Dershowitz and J.P. Jouannaud 1990; G. Huet 1980), an introduction to *occurrences* is given — a full development may be found in (G. Huet 1980).

Occurrences are sequences of natural numbers which are used to index the subterms of a term and their analysis forms the technical core of this paper. Let  $\mathcal{N}^*$  be the set of sequences of natural numbers with the empty sequence denoted  $\epsilon$ , while  $u \cdot v$  denotes the concatenation of u with v. If  $u \neq \epsilon$ , then  $u^+$  is the sequence obtained by omitting the last element of u, while  $u^-$  is the sequence obtained by omitting the first element. The prefix partial ordering is defined  $u \leq v$  iff  $\exists w.v = u \cdot w$  and in such a case define  $(u \cdot w)/u = w$ . These operations on sequences are extended pointwise to sets of sequences, e.g.  $X/u = \{w \mid u \cdot w \in X\}$ .

Now let  $\mathcal{T}$  be the terms of some calculus. Given any  $t \in \mathcal{T}$ , its set of *occurrences* is denoted  $O(t) \subseteq \mathcal{N}^*$ , while the subterm indexed at occurrence  $\sigma \in O(t)$  is denoted  $t/\sigma$ . These are defined as follows:

- If t is a variable, then 
$$O(t) = \{\epsilon\}$$
 and  $t/\epsilon = t$   
- If  $t = F(t_0, \dots, t_n)$ , then  $O(t) = \{\epsilon\} \cup \{i \cdot \sigma | i \le n, \sigma \in O(t_i)\}$  and  
 $t/\sigma = \begin{cases} t & \text{if } \sigma = \epsilon \\ t_i/\sigma^- & \text{if } \sigma \ne \epsilon \text{ and } \sigma = i \cdot \sigma^- \end{cases}$ 

When no danger of confusion exists, the distinction between an occurrence and the subterm so indexed may be blurred. As we shall see later, the conversion relation is not left linear in that different occurrences in the redex, which index syntactically equal subterms, may be mapped to the same occurrence in the reduct. To formalise this, a set X of occurrences is said to be *consistent* iff given any members  $\sigma, \sigma'$  of X, then  $t/\sigma = t/\sigma'$ , and if X is non-empty, the subterm so indexed is denoted t/X. Finally  $t[\sigma_i \leftarrow u_i]_{i \in \mathcal{I}}$  denotes the textual replacement of terms  $u_i$  at occurrences  $\sigma_i$  and is defined as expected (G. Huet 1980).

Given a rewrite relation R, if there is a rewrite  $t \Rightarrow_R t'$  we call t the redex and t' the reduct. The set of one-step R-reducts of a term t is denoted  $t/R = \{t' \mid (t,t') \in R\}$ , the reflexive closure of R is denoted  $R^+$  and the reflexive transitive closure of R is denoted  $R^+$  and the reflexive transitive closure of R is denoted  $R^+$  and the reflexive transitive closure of R is denoted  $R^+$  and the reflexive transitive closure of R is denoted  $R^+$  and the reflexive transitive closure of R is denoted  $R^+$  and the reflexive transitive closure of R is denoted  $R^+$  and the reflexive transitive closure of R is denoted  $R^+$  and the reflexive transitive closure of R is denoted  $R^+$  and the reflexive transitive closure of R is denoted  $R^+$  and the reflexive transitive closure of R is denoted t and t' are related in this theory we say t and t' are R-equivalent and write  $t =_R t'$ . If R is an equivalence relation, the equivalence class of an element t is denoted  $[t]_R$ , while if a term t is R-strongly normalising, its R-rank is denoted  $|t|_R$ . Some rewrite relations do not have normal forms and in these circumstances we use the more categorical notion of a quasi-normal form. A term t is an

| Table 1. Term Judgements of <b>ABCC</b>   |                                                                         |
|-------------------------------------------|-------------------------------------------------------------------------|
| $x\in \texttt{Var}(A)\cup\texttt{Con}(A)$ |                                                                         |
|                                           |                                                                         |
| $\frac{e:A=e':B}{(e,e'):A\times B}$       | $\frac{t:A_1 \times A_2}{-t+A}$                                         |
| $t: A_i$                                  | $\pi_i \iota: A_i \ t: A_1 + A_2  u: C  v: C  x_i \in 	extsf{Var}(A_i)$ |
| $\overline{\mathbf{in}_i(t):A_1+A_2}$     | $\mathbf{case}(t, x_1.u, x_2.v) : C$                                    |
| $e:B$ $x\in \mathtt{Var}(A)$              | $e:A{ ightarrow}B$ $e':A$                                               |
| $\lambda x.e:A{ ightarrow}B$              | e  e'  : B                                                              |

*R*-quasi-normal form iff whenever  $t \Rightarrow_R^* t'$ , then there is reduction sequence  $t' \Rightarrow_R^* t$  and the set of *R*-quasi-normal reducts of a term *t* is denoted R(t).

## 3. Almost Bicartesian Closed Logic

Although this paper is primarily concerned with the definition and decidability of  $\beta\eta$ equality for coproducts, in order to maintain continuity with previous work and to avoid certain trivial simplifications, a calculus which includes products, terminal object and exponentials is studied. This calculus is called "Almost Bicartesian Closed" as it corresponds to the internal language of bicartesian closed categories, without an initial object. We have verified separately that the techniques developed here are sufficient to cope with the addition of an initial object.

The *types* of "Almost Bicartesian Logic", denoted **ABCC**, are freely generated by the syntax

$$T := B \mid 1 \mid T + T \mid T \rightarrow T \mid T \times T$$

where B is any base type. For each type T, there are constants Con(T), including the special constant  $* \in Con(1)$ , and an infinite set of variables Var(T) such that if  $T \neq T'$ , then Var(T) and Var(T') are disjoint. This explicit typing of variables means contexts are not required to assign a type to a term and so the *term judgements* of **ABCC** are taken to be of the form t: T. These judgements are generated by the inference rules of Table 1.

Familiarity with calculi such as that above is assumed (J. Y. Girard *et al.* 1989; H. Barendregt 1984). Given any term judgement t: T, we say t is a term of type T. The free variables of a term t are denoted FV(t) and substitution of terms for free variables of the same type is defined as expected. A term is called an *introduction* term if it is a  $\lambda$ -abstraction, pair, injection or the constant \*. If a term is not an introduction term, then it is a *neutral* term. An occurrence  $\sigma \in O(t)$  is *negative* iff the subterm so indexed is either applied to another subterm, projected or the first argument of a *case*-expression — occurrences which are not negative are said to be *positive*.

**Lemma 3.1.** If there are typing judgements t: T and t: T', then T = T'.

*Proof.* Induction on the typing derivations.

The redex of the proposed  $\eta$ -rewrite rule for the sum type is expressed as a substitution and this may be formalised in terms of occurrences. Firstly, the variables bound at an occurrence  $\sigma \in O(t)$  are defined as follows:

$$\mathsf{BV}(\sigma,t) = \begin{cases} \emptyset & \text{if } \sigma = \epsilon \\ \{x\} \cup \mathsf{BV}(\sigma^-,t') & \text{if } t = \lambda x . t' \text{ and } \sigma \neq \epsilon \\ \{x_1\} \cup \mathsf{BV}(\sigma^-,t') & \text{if } t = \operatorname{case}(u, x_1.v_1, x_2.v_2) \text{ and } \sigma \ge 1 \\ \{x_2\} \cup \mathsf{BV}(\sigma^-,t') & \text{if } t = \operatorname{case}(u, x_1.v_1, x_2.v_2) \text{ and } \sigma \ge 2 \\ \mathsf{BV}(\sigma^-,t/i) & \text{otherwise }, \sigma = i \cdot \sigma^- \end{cases}$$

and the free occurrences of a term are  $FO(t) = \{\sigma \in O(t) \mid FV(t/\sigma) \cap BV(\sigma, t) = \emptyset\}$ . One easily proves by induction that if  $X \subseteq FO(t)$  is a non-empty, consistent set of free occurrences then, given a fresh variable  $z, t = t[\sigma \leftarrow z]_{\sigma \in X}[z := (t/X)]$ .

#### 4. A Rewrite Relation for ABCC

In (C. B. Jay and N. Ghani 1995) extensional rewrite relations for the product, unit and exponential were derived by constructing categorical models of reduction and taking introduction and elimination to be (locally) adjoint functors. When applied to coproducts this approach again generates a contractive  $\beta$ -rewrite rule and an expansionary  $\eta$ -rewrite rule.

To see this, let  $\mathcal{C}(X)$  is the category whose objects are terms of type X and whose morphisms are rewrites between terms. Assuming the variables x, y and z have the right type, the introduction and elimination rules for the coproduct, once extended to rewrites, form functors between the categories displayed in equation 3. When these functors are taken to constitute an adjoint pair

$$\mathcal{C}(C) \times \mathcal{C}(C) \xrightarrow{\mathbf{case}(z, x, \underline{a}, y, \underline{a})} \mathcal{C}(C) \xrightarrow{(-[\mathbf{in}_1(x)/z], -[\mathbf{in}_2(y)/z])} \mathcal{C}(C)$$
(3)

the associated unit and counit form the following expansionary  $\eta$ -rewrite rule and contractive  $\beta$ -rewrite rules.

These reduction rules, when closed under substitution and taken together with the reduction rules for the exponential, product and unit connectives, generate *expansionary* rewrite relation, denoted  $\Rightarrow$ , which is defined in Table 2. The fresh variables are assumed to have appropriate types and the capture of free variables is avoided by assuming  $x, y \notin FV(t)$  in the rewrite rule  $\eta_+$  and  $x \notin FV(t)$  in  $\eta_{\rightarrow}$ .

**Lemma 4.1.** If there is a typing judgement t : T and a rewrite  $t \Rightarrow t'$ , then there is a typing judgement t' : T.

*Proof.* The proof is by induction on the rewrite  $t \Rightarrow t'$ .

| Table 2.            | The Expansionary Re                         | write         | e Relation                                                           |                            |
|---------------------|---------------------------------------------|---------------|----------------------------------------------------------------------|----------------------------|
| $\beta_{\times,1}$  | $\pi_0 \langle a, b \rangle$                | $\Rightarrow$ | a                                                                    |                            |
| $\beta_{X,2}$       | $\pi_1 \langle a, b \rangle$                | $\Rightarrow$ | Ь                                                                    |                            |
| $\eta_{\times}$     | с                                           | $\Rightarrow$ | $\langle \pi_0 c, \pi_1 c \rangle$                                   | $\text{if } c: A \times B$ |
| $\beta \rightarrow$ | $(\lambda x.t)u$                            | $\Rightarrow$ | t[u/x]                                                               |                            |
| $\eta \rightarrow$  | t                                           | $\Rightarrow$ | $\lambda x.tx$                                                       | if $t : A \rightarrow B$   |
| $\eta_1$            | a                                           | $\Rightarrow$ | *                                                                    | if <i>a</i> :1             |
| $\beta_{+,1}$       | $\mathbf{case}(\mathbf{in}_1(t), x.u, y.v)$ | $\Rightarrow$ | u[t/x]                                                               |                            |
| $\beta_{+,2}$       | $\mathbf{case}(\mathbf{in}_2(t), x.u, y.v)$ | $\Rightarrow$ | v[t/y]                                                               |                            |
| $\eta_+$            | t[u/z]                                      | $\Rightarrow$ | $\mathbf{case}(u, x.t[\mathbf{in}_1(x)/z], y.t[\mathbf{in}_2(y)/z])$ | if $u: A + B$              |

The equational theory generated by the expansionary rewrite relation is called  $\beta\eta$ -equality and matches that suggested by the traditional categorical semantics for **ABCC**.

**Lemma 4.2.**  $\beta\eta$ -equality is sound and complete for models of **ABCC** in cartesian closed categories with coproducts.

*Proof.* Soundness is by induction on the term structure, while completeness follows from the construction of a free model, e.g. a category C whose objects are types and whose morphisms are  $\beta\eta$ -equivalence classes of terms:

$$\mathcal{C}(X,Y) = \{ [t]_{\beta\eta} \mid t : X \to Y \}$$

The rest of the proof follows the standard techniques, e.g. see (J.Lambek and P.Scott).  $\hfill \Box$ 

The  $\eta_+$ -rewrite rule is highly non-local in that consistent sets of free conversions may be expanded to the head of the term and is thus significantly more complex than the  $\eta$ -rewrite rules for the exponential and product. As terms typically contain many such sets of subterms, unique normal forms cannot be associated to terms; rather each term has a set of quasi-normal reducts, one for each of the different permutations in which subterms may be expanded. For example, the term  $\langle case(t, x.x, y.y), case(t', x'.x', y'.y') \rangle$ has two normal forms

$$\operatorname{case}(t, x.\operatorname{case}(t', x'.\langle x, x' \rangle, y'.\langle x, y' \rangle), y.\operatorname{case}(t', x'.\langle y, x' \rangle, y'.\langle y, y' \rangle))$$
(4)

 $\operatorname{and}$ 

$$\mathbf{case}(t', x'.\mathbf{case}(t, x.\langle x', x \rangle, y.\langle x', y \rangle), y'.\mathbf{case}(t, x.\langle y', x \rangle, y.\langle y', y \rangle))$$
(5)

depending on the order in which the subterms t and t' are expanded. To accommodate this feature, the  $\eta_+$ -rewrite rule is decomposed into two parts:

— The following special case of the  $\eta_+$ -rewrite rule is obtained by setting t to be the variable z in the definition of  $\eta_+$  given in Table 2.

$$u \Rightarrow \operatorname{case}(u, x.\operatorname{in}_1(x), y.\operatorname{in}_2(y))$$
 if  $u: A + B$  (6)

This rewrite rule is similar to the other  $\eta$ -rewrite rules of Table 2 in that terms of sum type are converted into negatively occurring subterms of the reduct. Indeed, once suitable restrictions have been imposed upon the applicability of the expansions in equation 6, and when taken together with the  $\beta$ -redexes and commuting conversions, the resulting rewrite relation is strongly normalising and confluent. The proof

of normalisation is essentially an adaptation of that in (C. B. Jay and N. Ghani 1995), although a couple of innovations are required to cope with some new technical problems.

— The second part of the decomposition is a generalisation of the "commuting conversions" appearing in (D. Prawitz 1971; J. Y. Girard *et al.* 1989). A conversion is a negative occurrence of sum type, or equivalently, the first argument of a case-expression. The conversion relation develops an algebra of these conversions, allowing them to be identified, discarded or expanded to the head of a term, e.g. the two normal forms in equations 4 and 5 are interconvertable in the conversion relation. Although not strongly normalising, each term has a (finite, enumerable) set of quasi-normal reducts and terms equivalent in the equational theory generated by the conversion relation have the same set of quasi-normal reducts. Confluence and decidability of the conversion relation are corollaries to these results.

Finally the whole expansionary rewrite relation is shown confluent and decidable by embedding it into the conversion relation. As the conversion relation is the main technical innovation in this paper, it is here that we begin.

#### 5. The Conversion Relation

The  $\eta_+$ -rewrite rule of Table 2 extracts consistent sets of free occurrences and inserts injections at their occurrences in the redex; when these occurrences are negative, new  $\beta_+$ -redexes are created by this process. The conversion relation restricts the  $\eta_+$ -rewrite rule to extract only negative occurrences, and then contracts these resulting  $\beta$ -redexes. This idea is formalised by (i) defining the set of conversions of a term; (ii) giving a recursive definition of the result of contracting the  $\beta$ -redexes mentioned above; and (iii) giving a calculus for deriving the rewrites of the conversion relation.

The set of *conversions* of a term t is defined by

 $C(t) = \{ \sigma \in D(t) | \sigma \text{ is a negative occurrence of sum type } \}$ 

The free conversions of t are simply those occurrences which are both free and conversions, i.e.  $FC(t) = C(t) \cap FO(t)$ . Every conversion has a *binding* which consists of the pair of variables bound by the arms of the *case*-expression associated to the conversion, e.g. the binding of the conversion 0 in the term case(t, x.u, y.v) consists of the pair x and y. These variable bindings play an important role in avoiding variable capture and henceforth whenever sets of conversions are considered, the subterms so indexed are assumed to have the same type and have the same binding.

Given a set  $X \subseteq C(t)$  of conversions, the result of contracting the  $\beta$ -redexes formed upon insertion of left injections at these occurrences is called the *first residue*, denoted  $t \setminus_1 X$ , while the result of contracting the  $\beta$ -redexes formed upon insertion of right injections is called the *second residue* and is denoted  $t \setminus_2 X$ . These terms are recursively defined as

| Table 3. The Conversion 1  | Relation                                                                                                                              |
|----------------------------|---------------------------------------------------------------------------------------------------------------------------------------|
|                            | $X \subseteq FC(t)  X \text{ consistent}  X \neq \emptyset$                                                                           |
| $\operatorname{Expansion}$ | $\langle \epsilon, X \rangle : t \Rightarrow_c \mathbf{case}(t/X, x.t \setminus_1 X, y.t \setminus_2 X)$                              |
|                            | $x \notin FV(t)  y \notin FV(t)$                                                                                                      |
| Weakening                  | $\langle \epsilon, \emptyset \rangle : \mathbf{case}(u, x.t, y.t) \Rightarrow_c t$                                                    |
|                            | $\langle \sigma, X \rangle : t_j \Rightarrow_c t'_j$                                                                                  |
| Congruence                 | $\overline{\langle j.\sigma, X \rangle} : \mathcal{T}(t_0, \dots, t_n) \Rightarrow_c \mathcal{T}(t_0, \dots, t_n)[j \leftarrow t'_j]$ |

follows:

$$t \setminus_i X = \begin{cases} t & \text{if } X = \emptyset \\ v_i \setminus_i X_i & \text{if } 0 \in X \text{ and } t = \mathbf{case}(u, x.v_1, y.v_2) \\ F(t_1 \setminus_i X_1, \cdots, t_n \setminus_i X_n) & \text{if } X \neq \emptyset, \ 0 \notin X \text{ and } t = F(t_1, \cdots, t_n) \end{cases}$$

where i = 1 or 2 and  $X_n = X/n$ .

**Lemma 5.1.** Given a set of conversions  $X \subseteq C(t)$  binding the variables  $x_1$  and  $x_2$ , then for i = 1 or 2 there is a reduction sequence  $t[\sigma \leftarrow \mathbf{in}_i(x_i)]_{\sigma \in X} \Rightarrow^* t \setminus_i X$ .

*Proof.* Induction over t.

The conversion relation is defined via a series of inference rules for deriving triples of the form  $\langle \sigma, X \rangle : t \Rightarrow_c t'$  where  $\sigma$  is the occurrence at which the actual redex occurs and X is the set of conversions to be expanded, i.e.  $X \subseteq FC(t/\sigma)$ . We call  $\langle \sigma, X \rangle$  the *label* of the rewrite, and, when not required, the label part of the rewrite is omitted. These triples are generated by the inference rules of Table 3, where in the *Expansion* clause x, y are the variables bound by each  $\sigma \in X$  and to avoid variable capture we assume  $x, y \notin FV(t) \cup BV(\sigma, t)$ . These conditions can always be met, if necessary by a change of bound variables.

The Expansion clause requires the set X of conversions to be free and consistent so that the redex may be expressed as a substitution and hence in a form compatible with the  $\eta_+$ -rewrite rule of Table 2. In addition, this set is required to be non-empty to prevent expansions of the form  $u \Rightarrow case(t, x.u, y.u)$  which would allow terms to grow arbitrary large, new free variables to be introduced and other undesirable features. However these terms remain identified in the equational theory generated by the conversion relation because redex and reduct have been inverted and included under the Weakening clause.

**Lemma 5.2.** Given a triple  $\langle \sigma, X \rangle$  :  $t \Rightarrow_c t'$ , then t = t' in the expansionary rewrite relation.

*Proof.* The lemma is proved by induction on  $\sigma$ . If  $\sigma = \epsilon$  and the rewrite is of the form  $case(u, x.t, y.t) \Rightarrow_c t$  then, given a variable z not free in t

 $t = t[u/z] \Rightarrow_{\eta_{+}} \operatorname{case}(u, x.t[\operatorname{in}_{1}(x)/z], y.t[\operatorname{in}_{2}(y)/z]) = \operatorname{case}(u, x.t, y.t)$ 

However, if X in non-empty then

 $t = t[\sigma \leftarrow z]_{\sigma \in X} [z := t/X]$ 

$$\begin{aligned} &\Rightarrow_{\eta_{+}} \quad \mathbf{case}(t/X, x.t[\sigma \leftarrow \mathbf{in}_{1}(x)]_{\sigma \in X}, y.t[\sigma \leftarrow \mathbf{in}_{2}(y)]_{\sigma \in X}) \\ &\Rightarrow^{*} \quad \mathbf{case}(t/X, x.t \setminus_{1} X, y.t \setminus_{2} X) \end{aligned}$$

where the equality in the first line holds because X is a non-empty, consistent set of free conversions, and the last line is by lemma 5.1. Finally if  $\sigma \neq \epsilon$  then, as both relations are congruences, the lemma follows by induction.

The conversion relation is so named because the relation generalises the commuting conversions occurring in the literature (J. Y. Girard *et al.* 1989; D. Prawitz 1971). Commuting conversions are formed when negative occurrences index case-expressions — an example is given in in equation 7. The reader is invited to check that this rewrite may be derived as a conversion rewrite with label  $\langle \epsilon, \{00\} \rangle$ .

$$\mu: \quad \operatorname{case}(\operatorname{case}(t, x.u, y.v), x'.u', y'.v') \Rightarrow$$

$$\operatorname{case}(t, x.\operatorname{case}(u, x'.u', y'.v'), y.\operatorname{case}(v, x'.u', y'.v'))$$

$$(7)$$

The rewrite relation  $\Rightarrow_{\mu}$  is defined to be the least congruence containing the redex given in equation 7. Note that  $\Rightarrow_{\mu}$  is strongly normalising and confluent and so has unique normal forms.

The technical core of the analysis of the conversion relation uses the structure of a rewrite  $r: t \Rightarrow_c t'$ , represented in the label r, to define a relation  $\overline{r} \subseteq C(t) \times C(t')$  which relates conversions in the redex, called *ancestors*, to conversions in the reduct, called *descendants*. As we have seen, the action of a (consistent) set of conversions  $X \subseteq C(t)$  is to produce two residues, namely  $t \setminus_1 X$  and  $t \setminus_2 X$ . This action induces a partitioning of the set of conversions C(t) into (i) those conversions which are sub-conversions of (unique) members of X; (ii) those conversions which have descendants in one or both of the residues; and (iii) those conversions which fit into neither of these categories and hence have no descendants.

If a conversion  $\sigma$  has a descendant in the residue  $t \setminus_i X$ , then this descendant will be unique and is given by the partial function  $\Omega_i(X, \sigma)$  (which is undefined if  $\sigma$  has no descendant).

$$\Omega_i(X,\sigma) = \begin{cases} \Omega_i(X/i,\sigma^-) & \text{if } 0 \in X \text{ and } \sigma \ge i \\ \text{undefined} & \text{if } 0 \in X \text{ and } \sigma \ge i \\ 0 & \text{if } 0 \notin X \text{ and } \sigma = 0 \\ i.\Omega_i(X/i,\sigma^-) & \text{otherwise }, \sigma = i \cdot \sigma^- \end{cases}$$

Note that in general the domain of  $\Omega_1$  will differ from  $\Omega_2$ .

**Lemma 5.3.** Let  $X \subseteq C(t)$  and  $\tau \in C(t)$  be in the domain of  $\Omega_i(X)$ . Then  $\Omega_i(X, \tau) \in C(t \setminus X)$  is a conversion, indexing the subterm

$$(t \setminus_i X) / \Omega_i(X, \tau) = (t/\tau) \setminus_i (X/\tau)$$

In addition the partial function  $\Omega_i(X) : C(t) \rightarrow C(t \setminus iX)$  is surjective, injective on its domain, maps free conversions to free conversions and is strictly monotonic, i.e. for conversions  $\sigma, \tau$  in its domain:

$$\sigma > \tau$$
 iff  $\Omega_i(X, \sigma) > \Omega_i(X, \tau)$ 

*Proof.* The proof is by induction on the definition of  $\Omega_i$ .

The conversion tracking function promised at the beginning of this section can now be constructed. Given a rewrite  $r: t \Rightarrow_c t'$ , and a conversion  $\sigma \in C(t)$ , define its set of descendants  $\overline{r}(\sigma) \subseteq C(t')$  as follows:

— If r is a Weakening, then

$$\overline{r}(\tau) = \begin{cases} \emptyset & \text{if } \tau \ge 0\\ \{\tau^-\} & \text{otherwise} \end{cases}$$

— If r is an *Expansion* of the non-empty set of conversions X, then

$$\overline{r}(\tau) = \begin{cases} \{0.\tau/\sigma\} & \text{if there is a } \sigma \in X \text{ with } \tau \ge \sigma \\ \{1.\Omega_1(X,\tau), 2.\Omega_2(X,\tau)\} & \text{otherwise} \end{cases}$$

where, since X is consistent,  $\sigma$  in the first clause is necessarily unique and those functions undefined in the second clause are deleted.

— If r is induced by a congruence then

$$\overline{\langle j \cdot \omega, X \rangle}(\sigma) = \begin{cases} \{\sigma\} & \text{if } \sigma = 0 \text{ or } \sigma \neq j \cdot \sigma^{-1} \\ j \cdot \overline{\langle \omega, X/j \rangle}(\sigma^{-1}) & \text{otherwise} \end{cases}$$

The function  $\overline{r}$  is extended pointwise to sets of conversions. Note that a single conversion may have more than one descendant and the ordering on conversions is not necessarily preserved. Both of these points can be seen in equation 7 where  $\overline{r}(00) = \{0\}$  while  $\overline{r}(0) = \{10, 20\}$ . Another interesting reduction is

$$\langle \epsilon, \{0, 10\} \rangle : \mathbf{case}(t, x.\mathbf{case}(t, x.u, y.v), y.s) \Rightarrow \mathbf{case}(t, x.u, y.s)$$
(8)

which shows how a conversion, e.g. any inside v in the redex, may have no descendants, and how a conversion in the reduct may have more than one ancestor. However,  $\overline{r}$  is surjective, i.e. all conversions in a reduct have at least one ancestor in the redex. Thus the possibilities for further rewriting, which are determined by the conversions of the reduct, may be traced back to the associated redex, and hence we may directly construct the quasi-normal reducts of a term and deduce confluence and decidability.

**Lemma 5.4.** Given a rewrite  $r: t \Rightarrow_c t'$ , and a conversion  $\sigma \in C(t)$ , then  $\overline{r}(\sigma) \subseteq C(t')$  and the set

$$r^{-1}(\tau) = \{ \sigma \in \mathsf{C}(t) \mid \tau \in \overline{r}(\sigma) \}$$

is non-empty.

Proof. Induction on the rewrite.

#### 6. A Decidability Result

Given a rewrite  $r: t \Rightarrow_c t'$  and a set of conversions  $X \subseteq C(t)$ , we give conditions under which r may be localised to its action on an individual conversion and to its action on the residues, i.e. find conditions under which rewrites of the following form exist

$$r/\sigma': t/\sigma \Rightarrow_c t'/\sigma'$$
 and  $r \setminus_i X: t \setminus_i X \Rightarrow_c t' \setminus_i \overline{r}(X)$ 

where  $\sigma \in X$  and  $\sigma' \in \overline{r}(\sigma)$ . As these localised rewrites have smaller redexes, we obtain a recursive decomposition of the conversion relation which will be the key to the construction of quasi-normal forms.

Equation 7 shows that a rewrite cannot always be localised to a conversion  $-\overline{r}(0) = \{10, 20\}$ , but no rewrite exists between the corresponding subterms. This problem occurs as the conversion 0 is mapped into the residues while one of its sub-conversions is expanded to the head of the term and is thus 'removed' from the original conversion. The key to localising a rewrite to a conversion lies in ensuring that the layer structure on conversions, formed by the embedding of conversions inside each other, is preserved. A rewrite  $\langle \tau, X \rangle : t \Rightarrow t'$  is said to *preserve* a conversion  $\sigma \in C(t)$  iff  $\forall \omega \in X. \neg (\tau < \sigma < \omega)$ , i.e. no subconversions of  $\sigma$  are expanded outside of  $\sigma$ . This generates a subrelation of the conversion relation where all conversions are preserved.

$$\Rightarrow_{p} = \{r : t \Rightarrow_{c} t' \mid r \text{ preserves } C(t)\}$$

As mentioned above, the commuting conversion in equation 7 is an example of a rewrite which does not preserve the conversion layer structure. In fact, this redex fully describes all the cases in which  $\Rightarrow_c$ -reduction fails to preserve a conversion.

Lemma 6.1. The conversion relation may be decomposed as follows:

$$\Rightarrow_c^* = (\Rightarrow_p \cup \Rightarrow_\mu)^*$$

*Proof.* Induction on the structure of a conversion rewrite.

**Lemma 6.2.** Let  $r : t \Rightarrow_c t'$  preserve  $\sigma \in C(t)$ . Then for all  $\sigma' \in \overline{r}(\sigma)$  there is a rewrite  $r/\sigma : t/\sigma \Rightarrow_c t'/\sigma'$ . In addition, if r preserves all conversions in C(t), then  $r/\sigma$  preserves all conversions in  $C(t/\sigma)$ .

*Proof.* Induction on r.

We now consider the conditions which must be satisfied by a set of conversions  $X \subseteq C(t)$ such that a rewrite  $r : t \Rightarrow_c t'$  induces rewrites of the residues  $r \setminus_i X : t \setminus_i X \Rightarrow_c t' \setminus_i \overline{r}(X)$ . If r is of the form  $case(t, x.u, y.u) \Rightarrow_c u$ , and X contains a conversion inside one of the subterms u, then in order to maintain the shape of the redex, X must also contain the "sister" conversion inside the other arm. Similar considerations apply if r is an expansion  $\langle \epsilon, Y \rangle$ , where if X contains a conversion occurring inside an element of Y then, in order to maintain consistency, X must contain the sister conversions inside the other elements of Y.

These conditions are easily formalised in terms of the conversion tracking function. Given a rewrite r, a set of conversions X is r-closed iff  $r^{-1}\overline{r}(X) = X$ .

**Lemma 6.3.** Let  $r: t \Rightarrow_c t' X$  be an *r*-closed set of conversions. Then there are either rewrites  $r \setminus_i X : t \setminus_i X \Rightarrow_c t' \setminus_i \overline{r}(X)$  or rewrites in the reverse direction. In addition, if *r* preserves all conversions then so do the residual rewrites.

*Proof.* The proof is by induction on r.

An example of the need to reverse the direction of the residual rewrite is the following. If r represents an expansion of conversions  $Y \subseteq C(t)$ , then the obvious candidate for  $r \setminus_i X$  is the basic expansion of the descendants of Y in the residue  $t \setminus_i X$ , namely  $\Omega_i(X, Y)$ .

However this set may be empty, in which case the direction of the residual rewrite must be reversed and taken to be a *Weakening*.

Equivalence in the conversion relation can now be proved decidable but, because of the conditions required by *preservation* and *closure*, this is done firstly for the subrelation  $\Rightarrow_p$ . For each term, we construct its finite set of  $\Rightarrow_p$ -quasi normal forms and show that  $\Rightarrow_p$ -equivalent terms have the same set of  $\Rightarrow_p$ -quasi-normal forms.

If a term is a  $\Rightarrow_p$ -quasi-normal form containing free conversions, then the term must be a case-expression, as otherwise a rewrite to such a term would exist, but not one in the other direction. Thus the construction of  $\Rightarrow_p$ -quasi-normal forms is essentially a process of expanding as many conversions as possible. However as a non-free conversion may have a free  $\Rightarrow_p$ -descendant, we must consider not just free conversions but also *potentially* free conversions, and secondly, as these quasi-normal forms are to be  $\Rightarrow_p$ -reducts, only minimal conversions are expanded. The construction of  $\Rightarrow_p$ -quasi-normal forms also performs two other tasks, namely checking for possible applications of Weakening and also ensuring that as much identification of conversions occurs as is possible.

The construction of  $\Rightarrow_p$ -quasi-normal forms is presented in Table 4 in terms of a function  $NF_p$  which maps a term to its set of  $\Rightarrow_p$ -quasi-normal reducts. In this table, the following definitions are used:

— The set of *minimal conversions* of a term are given by:

$$\mathsf{MC}(t) = \{ \sigma \in \mathsf{C}(t) \mid \not\exists \sigma' \in \mathsf{C}(t) . \sigma' < \sigma \}$$

— The set of *potentially free conversions* of a term are given by:

$$\mathsf{PFC}(t) = \{ \sigma \!\in\! \mathsf{C}(t) \mid \forall u \!\in\! \mathsf{NF}_p(t/\sigma) . \mathsf{BV}(\sigma,t) \cap \mathsf{FV}(u) = \emptyset \}$$

- The set of *potentially free conversions* of a term are  $MPFC(t) = PFC(t) \cap MC(t)$ .
- MPFC(t) is equipped with an equivalence relation determining which conversions are to be identified:

$$\sigma_1 \sim \sigma_2$$
 iff  $NF_p(t/\sigma_1) = NF_p(t/\sigma_2)$ 

Note that for any  $\sigma \in \mathbf{C}(t)$ ,  $\sigma \neq \epsilon$  and hence the size of the term  $t/\sigma$  is less than the size of t. Hence  $\mathbf{PFC}(t)$ , which is defined in terms of  $\mathbf{NF}_p(t/\sigma)$ , is well defined. The set of terms  $\mathbf{NF}_p(t)$  is clearly non-empty, finite, enumerable and the minimality condition ensures that if  $t' \in \mathbf{NF}_p(t)$  then  $t \Rightarrow_p^* t'$ .

**Lemma 6.4.** Let  $r: t \Rightarrow_p t'$ .

- If  $\sigma \in C(t)$  and  $\sigma' \in \overline{r}(\sigma)$ , then  $\sigma \in MPFC(t)$  iff  $\sigma' \in MPFC(t')$  and for such a minimal potentially free conversion  $\sigma, [\sigma]_{\sim}$  is r-closed and  $\overline{r}([\sigma]_{\sim}) = [\sigma']_{\sim}$ . The sets NF(t) and NF(t') are equal

— The sets  $NF_p(t)$  and  $NF_p(t')$  are equal.

*Proof.* The lemma is proved simultaneously by induction on the sum of the sizes of the terms in question. That  $\sigma$  is minimal iff  $\sigma'$  is follows by induction on the definition of the function  $\overline{r}$  and the fact that r preserves all conversions. By lemma 6.2 there is a rewrite  $t/\sigma \Rightarrow_p t'/\sigma'$  and so by the induction hypothesis  $NF_p(t/\sigma) = NF_p(t'/\sigma')$ . Thus for

Table 4. The Function  $NF_p$ — If t is a variable then  $NF_p(t) = \{t\}$ . — If t is not a variable and  $MPFC(t) = \emptyset$ , then  $\frac{t = \mathcal{T}(t_0, \dots, t_n) \quad \alpha_i \in NF_p(t_i)}{\mathcal{T}(\alpha_0, \dots, \alpha_n) \in NF_p(t)}$ — If t is not a variable and  $\sigma \in MPFC(t)$ , then either  $\frac{NF_p(t \setminus 1[\sigma]_{\sim}) = NF_p(t \setminus 2[\sigma]_{\sim}) \quad u \in NF_p(t \setminus 1[\sigma]_{\sim})}{u \in NF_p(t \setminus 1[\sigma]_{\sim})}$ or  $\frac{NF_p(t \setminus 1[\sigma]_{\sim}) \neq NF_p(t \setminus 2[\sigma]_{\sim}) \quad \beta_i \in NF_p(t \setminus i[\sigma]_{\sim}) \quad \alpha \in NF_p(t/\sigma)}{case(\alpha, x, \beta_1, y, \beta_2) \in NF_p(t)}$ where x, y are the variables bound by the set of conversions  $[\sigma]_{\sim}$ .

any element u of  $NF_p(t/\sigma)$ ,  $FV(u) \subseteq FV(t/\sigma) \cap FV(t'/\sigma')$  and hence

$$\begin{split} x \in \mathtt{BV}(\sigma, t) \cap \mathtt{FV}(u) & \text{iff} \quad x \in \mathtt{BV}(\sigma, t) \cap \mathtt{FV}(t'/\sigma') \\ & \text{iff} \quad x \in \mathtt{BV}(\sigma', t') \cap \mathtt{FV}(t'/\sigma') \\ & \text{iff} \quad x \in \mathtt{BV}(\sigma', t') \cap \mathtt{FV}(u) \end{split}$$

where the equality

$$\mathtt{BV}(\sigma,t) \cap \mathtt{FV}(t'/\sigma') = \mathtt{BV}(\sigma',t') \cap \mathtt{FV}(t'/\sigma')$$

may be proved by induction on the rewrite r. Thus  $\sigma$  is potentially free iff  $\sigma'$  is. Finally, given  $\tau \in r^{-1}\overline{r}[\sigma]_{\sim}$ , there is a  $\tau' \in \text{MPFC}(t')$  and a  $\omega \in [\sigma]_{\sim}$  such that  $t/\tau \Rightarrow_p t'/\tau'$  and  $t/\omega \Rightarrow_p t'/\tau'$ . Thus by the induction hypothesis

$$NF_p(t/\tau) = NF_p(t'/\tau') = NF_p(t/\omega) = NF_p(t/\sigma)$$

and so  $[\sigma]_{\sim}$  is r-closed. The equation  $\overline{r}([\sigma]_{\sim}) = [\sigma']_{\sim}$  may be proved similarly by direct calculation.

For the second half of the lemma there are two possibilities. Firstly if  $MPFC(t) = \emptyset$ , then by the first part of this lemma  $MPFC(t') = \emptyset$  and so the lemma follows by the induction hypothesis. If however there is a  $\sigma \in MPFC(t)$  then, because  $[\sigma]_{\sim}$  is *r*-closed, by lemmas 6.2 and 6.3 there are rewrites

$$t/\sigma \Rightarrow_p t'/\sigma'$$
 and  $t \setminus_1[\sigma]_{\sim} \Rightarrow_p t' \setminus_1 \overline{r}([\sigma]_{\sim})$  and  $t \setminus_2[\sigma]_{\sim} \Rightarrow_p t' \setminus_2 \overline{r}([\sigma]_{\sim})$ 

where  $\sigma' \in \overline{r}(\sigma)$  and the directions of the second and third reductions may be reversed. For each of these rewrites, the set of normal forms of the left hand side is the same as those of the right hand side. It is now routine to check that the sets of terms  $NF_p(t)$  and  $NF_p(t')$  are equal.

**Lemma 6.5.** The relation  $\Rightarrow_p$  is confluent and has a decidable equational theory, while if  $t' \in NF_p(t)$  then t' is a  $\Rightarrow_p$ -quasi-normal form.

*Proof.* By lemma 6.4 any  $\Rightarrow_p^*$ -span with redex t has a  $\Rightarrow_p^*$ -co-span to any element of  $NF_p(t)$ . Thus  $\Rightarrow_p$  is confluent while the associated equational theory may be decided by

| Table 5. Definition of NF and NF <sup>o</sup>                                                       |
|-----------------------------------------------------------------------------------------------------|
| — If $t$ is a variable                                                                              |
| $\overline{\mathtt{NF}^o(t)=\{t\}}$                                                                 |
| — If $t$ is a case-expression                                                                       |
| $\alpha \in \mathtt{NF}(u)  \beta_i \in \mathtt{NF}^o(v_i)$                                         |
| $\mu \mathbf{case}(\alpha, x.\beta_1, y.\beta_2) \in \mathbb{IF}^o(\mathbf{case}(u, x.v_1, y.v_2))$ |
| — If $t$ is not a case-expression or a variable                                                     |
| $t = \mathcal{T}(t_0, \dots, t_n)  \alpha_i \in \mathbf{WF}^o(t_i)$                                 |
| $\mathcal{T}(lpha_0,\ldots,lpha_n)\in \mathtt{NF}^o(t)$                                             |
| — $\mathbf{NF}(t)$ is defined                                                                       |
| $\mathbf{NF}(t) = \bigcup \mathbf{NF}_p(\alpha)$                                                    |
| $lpha\in \mathbb{N}\mathbf{F}^{o}\left( t ight)$                                                    |

comparing the quasi-normal forms just constructed. Finally, if  $\alpha \in NF_p(t)$  and  $\alpha \Rightarrow_p^* \alpha'$ , then by lemma 6.4  $NF_p(\alpha') = NF_p(t)$  and so  $\alpha' \Rightarrow_p^* \alpha$ . Thus  $\alpha$  is a  $\Rightarrow_p$ -quasi-normal form.

# 7. Decidability of $\Rightarrow_c$ -Equivalence

By lemma 6.1 any  $\Rightarrow_c$ -reduction can be expressed as a sequence of commuting conversions given in equation 7 and  $\Rightarrow_p$ -reductions. Thus the construction of  $\Rightarrow_c$ -quasi-normal forms is a process of combining the  $\Rightarrow_p$ -quasi-normal forms just defined with the normal forms of the commuting conversion relation. This is a three stage process which firstly recursively normalises all minimal conversions and then contracts all  $\mu$ -commuting conversions. Such terms are *stable* and only have  $\Rightarrow_p$ -reducts to other stable terms — hence the procedure is completed by using operator NF<sub>p</sub> defined in the last section.

Recall that  $\Rightarrow_{\mu}$  is the least congruence on terms containing the reductions in equation 7. This relation is well known to be strongly normalising (D. Prawitz 1971) and local confluence is easily shown — thus we may define  $\mu(t)$  to be the unique  $\Rightarrow_{\mu}$ -normal form of t. The functions **NF** and **NF**<sup>o</sup> which map terms to sets of terms are defined simultaneously in Table 5. An alternative definition of **NF**<sup>o</sup>(t), which will be used later, is the following:

$$NF^{o}(t) = \{ \mu(t[\sigma \leftarrow \alpha_{\sigma}]_{\sigma \in MC(t)}) \mid \alpha_{\sigma} \in NF(t/\sigma) \}$$

A term is *stable* iff  $\forall \sigma \in C(t).PFC(t/\sigma) = \emptyset$ . Stable terms are important as they are both  $\Rightarrow_{\mu}$ -normal forms and closed under  $\Rightarrow_{p}$ -reduction. Hence the  $\Rightarrow_{c}$ -quasi-normal reducts of a stable term t will be the set  $NF_{p}(t)$ .

**Lemma 7.1.** If  $r: t \Rightarrow_c t'$  and t is stable, then r preserves all conversions and t' is stable. In addition, if  $\alpha \in NF^{\circ}(t)$  then  $\alpha$  is stable and any  $\alpha' \in NF(t)$  is a  $\Rightarrow_c$ -quasi-normal form.

*Proof.* By stability no conversion in t contains a free subconversion and hence t must be a  $\Rightarrow_{\mu}$ -normal form. Thus any reduct of t must be a  $\Rightarrow_{p}$ -reduct. If there is a conversion in t' containing a potentially free sub-conversion, then there is also a conversion in t' containing a minimal potentially free sub-conversion. Thus by lemma 6.4 there is

a conversion in t also containing a minimal potentially free conversion and so the redex can't be stable.

That the function  $NF^{\circ}$  creates stable terms is proved by induction on the term structure, using the first half of this lemma to establish the result for case-expressions. Also by the first half of this lemma all  $\Rightarrow_c$ -reducts of  $\alpha'$  are actually  $\Rightarrow_p$ -reducts, and because  $\alpha'$  is a  $\Rightarrow -p$ -quasi-normal, there must be a reduction in the reverse direction. Thus  $\alpha'$  is a  $\Rightarrow_c$ -quasi-normal form.

Proving that  $t =_c t'$ , then  $NF^o(t) = NF^o(t')$  by explicitly considering each quasi-normal form of each term is too time consuming. A simpler approach is to show that there are  $\Rightarrow_p$ -equivalent members of  $NF^o(t)$  and  $NF^o(t')$  and the key is the following technical lemma which relates commuting conversions to  $\Rightarrow_c$ -reduction.

**Lemma 7.2.** If  $t \Rightarrow_c t'$ , then  $\mu(t) \Rightarrow_c^* \mu(t')$ .

*Proof.* The proof follows the same pattern as lemma 9.3. Note that the notion of a full parallel rewrite is modified to prevent only the introduction of new  $\mu$ -redexes. **Lemma 7.3.** Given a term t and two terms  $\alpha, \alpha' \in NF^{\circ}(t)$  then  $\alpha \Rightarrow_{p}^{*} \alpha'$ . Thus given terms  $\alpha \in NF^{\circ}(t)$  and  $\alpha' \in NF^{\circ}(t')$ , if  $\alpha =_{p} \alpha'$  then the sets NF(t) and NF(t') are equal.

*Proof.* The first part of the lemma is proved by induction on the term t with the only interesting part being if t is a case-expression, say case(t', x.u, y.v). Then  $\alpha$  and  $\alpha'$  are of the form

$$\alpha \equiv \mu(\mathbf{case}(\alpha_0, x.\beta_1, y.\beta_2)) \text{ and } \alpha' \equiv \mu(\mathbf{case}(\alpha'_0, x.\beta'_1, y.\beta'_2))$$

By the induction hypothesis  $\beta_i \Rightarrow_p^* \beta'_i$  and  $\alpha_0 =_p \alpha'_0$ , and since  $\alpha'_0$  is a  $\Rightarrow_p$ -quasi-normal form and  $\Rightarrow_p$  is confluent, there is a reduction sequence  $\alpha_0 \Rightarrow_p^* \alpha'_0$ . The lemma now follows from lemma 7.2.

For the second half of the lemma, given a term  $\alpha_0 \in NF(t)$ , there is a term  $\alpha_1 \in NF^o(t)$  such that  $\alpha_0 \in NF_p(\alpha_1)$ . By the first part of this lemma  $\alpha_1 =_p \alpha =_p \alpha'$  and so  $\alpha_0 \in NF(t')$ . Thus we have shown that  $NF(t) \subseteq NF(t')$  and as the argument is symmetric, the reverse containment also holds.

**Lemma 7.4.** The terms case(t, x.u, y.u) and u have the same set of normal forms.

*Proof.* If  $\alpha_0 \in NF(t)$  and  $\alpha_1 \in NF^o(u)$ , then by lemma 7.2

$$\mu(\mathbf{case}(\alpha_0, x.\alpha_1, y.\alpha_1)) \Rightarrow_c^* \mu(\alpha_1) = \alpha_1$$

The redex is a member of  $NF^{\circ}(\mathbf{case}(t, x.u, y.u))$ , and hence the reduction sequence is actually a  $\Rightarrow_p^*$ -reduction sequence. In addition the reduct is a member of  $NF^{\circ}(u)$  and so the lemma now follows from lemma 7.3.

Lemma 7.5. The terms

$$t_0 = \mathbf{case}(\mathbf{case}(t, x_1.u_1, x_2.u_2), y_1.v_1, y_2.v_2)$$

 $\operatorname{and}$ 

$$t_1 = \mathbf{case}(t, x_1.\mathbf{case}(u_1, y_1.v_1, y_2.v_2), x_2.\mathbf{case}(u_2, y_1.v_1, y_2.v_2))$$

have the same set of quasi-normal reducts.

*Proof.* Let  $\alpha \in NF^{\circ}(\mathbf{case}(t, x_1.u_1, x_2.u_2))$  and  $\alpha_{v_i} \in NF^{\circ}(v_i)$ . Then given an  $\alpha' \in NF_p(\alpha)$ , by lemma 7.2 there is a reduction

$$\mu(\mathbf{case}(\alpha, y_1.\alpha_{v_1}, y_2.\alpha_{v_2})) \Rightarrow^*_p \mu(\mathbf{case}(\alpha', y_1.\alpha_{v_1}, y_2.\alpha_{v_2}))$$

where the reduct is a member of  $NF^{o}(t_0)$ . Now  $\alpha$  must be of the form

$$\mu(\mathbf{case}(\alpha_t, x_1.\alpha_{u_1}, x_2.\alpha_{u_2}))$$

where  $\alpha_t \in NF(t)$  and  $\alpha_{u_i} \in NF^o(u_i)$ . Again by lemma 7.2, if  $\alpha'_{u_i} \in NF_p(\alpha_{u_i})$  then there is a  $\Rightarrow_p^*$ -reduction sequence:

$$\begin{aligned} \mu \mathbf{case}(\alpha, y_1 . \alpha_{v_1}, y_2 . \alpha_{v_2}) \\ &= \mu \mathbf{case}(\mathbf{case}(\alpha_t, x_1 . \alpha_{u_1}, x_2 . \alpha_{u_2}), y_1 . \alpha_{v_1}, y_2 . \alpha_{v_2}) \\ &\Rightarrow_p^* \quad \mu \mathbf{case}(\alpha_t, x_1 . \mathbf{case}(\alpha_{u_1}, y_1 . \alpha_{v_1}, y_2 . \alpha_{v_2}), x_2 . \mathbf{case}(\alpha_{u_2}, y_1 . \alpha_{v_1}, y_2 . \alpha_{v_2})) \\ &\Rightarrow_p^* \quad \mu \mathbf{case}(\alpha_t, x_1 . \mathbf{case}(\alpha'_{u_1}, y_1 . \alpha_{v_1}, y_2 . \alpha_{v_2}), x_2 . \mathbf{case}(\alpha'_{u_2}, y_1 . \alpha_{v_1}, y_2 . \alpha_{v_2})) \end{aligned}$$

As the reduct of this sequence is a member of  $NF^{o}(t_{1})$ , we have shown that there are  $\Rightarrow_{p}$ -equivalent members of  $NF^{o}(t_{0})$  and  $NF^{o}(t_{1})$ . Hence by lemma 7.3 the normal forms of  $t_{0}$  and  $t_{1}$  are equal.

**Lemma 7.6.** Let  $X \subseteq MC(t)$  be a non-empty consistent set of minimal free conversions. Then there is a term  $\alpha \in NF^{\circ}(t)$  and  $\alpha' \in NF^{\circ}(case(t/X, x.t \setminus X, y.t \setminus X))$  such that  $\alpha \Rightarrow_{p}^{*} \alpha'$ . Thus these terms have the same set of quasi-normal reducts.

*Proof.* Let  $\alpha_X \in NF(t/X)$  and define the term

$$t_0 = t[\sigma \leftarrow \alpha_\sigma]_{\sigma \in MC(t)}$$

where  $\alpha_{\sigma}$  are chosen members of  $NF(t/\sigma)$  such that if  $\sigma \in X$ ,  $\alpha_{\sigma} = \alpha_X$ . Then  $X \subseteq MC(t_0)$  is a set of minimal, free consistent conversions and so there is a rewrite

$$t_0 \Rightarrow_p \mathbf{case}(\alpha_X, x.t_0 \setminus X, y.t_0 \setminus X)$$

and hence by lemma 7.2 there is also a reduction sequence

$$\mu t_0 \Rightarrow_c^* \mu \mathbf{case}(\alpha_X, x.\mu(t_0 \setminus X), y.\mu(t_0 \setminus X))$$

Now for any conversion  $\tau \in C(t)$ ,  $\tau$  is minimal iff  $\Omega_i(X, \tau)$  (when defined) is minimal and, providing this is the case,  $(t \setminus X)/\Omega_i(X, \tau) = t/\tau$ . Thus

$$t_0 \setminus_i X = t[\sigma \leftarrow \alpha_\sigma]_{\sigma \in MC(t)} \setminus_i X$$
  
=  $(t \setminus_i X) [\Omega_i(X, \sigma) \leftarrow \alpha_\sigma]_{\sigma \in MC(t)}$   
=  $(t \setminus_i X) [\sigma' \leftarrow \alpha_{\sigma'}]_{\sigma' \in MC(t \setminus_i X)}$ 

and hence  $\mu(t_0 \setminus X) \in NF^o(t \setminus X)$ . Thus the term  $\mu(t_0)$  is a member of  $NF^o(t)$  and  $\Rightarrow_{p}$ -rewrites to a member of  $NF^o(\mathbf{case}(t/X, x.t \setminus X, y.t \setminus X))$  and so the lemma follows from lemma 7.3.

**Theorem 7.7.** If  $r:t \Rightarrow_c t'$  then NF(t) = NF(t'). Hence the conversion relation is confluent and conversion-equivalence decidable.

```
Table 6. The Rewrite Relation \Rightarrow_{\beta}
          \beta_{X,i}
                                           \pi_i \langle u_0, u_1 \rangle
                                                                ⇒
                                                                        u_i
                                                                        \mathbf{case}(t, x.\pi_i u, y.\pi_i v)
                              \pi_i \mathbf{case}(t, x.u, y.v)
          \beta_{+,\times}
                                                                \Rightarrow
                                               (\lambda x.t)t'
                                                                        t[t'/x]
                                                                \Rightarrow
                               case(t, x.u, y.v)t'
                                                                        case(t, x.ut', y.vt')
          \beta_{+,-}
                                                                ⇒
                          case(in_1(t), x.u, y.v)
          \beta_{+,1}
                                                                ⇒
                                                                        u[t/x]
                          case(in_2(t), x.u, y.v)
                                                                        v[t/y]
          \beta_{+,2}
                                                                ⇒
                          case(case(t, x.u, y.v), x'.u', y'.v') \Rightarrow
        \beta_{+,+} :
                                                                \operatorname{case}(t, x.\operatorname{case}(u, x'.u', y'.v'), y.\operatorname{case}(v, x'.u', y'.v'))
```

*Proof.* Induction on the label r of the rewrite  $r:t \Rightarrow_c t'$  is used to show that  $NF^o(t)$  and  $NF^o(t')$  have  $\Rightarrow_p$ -equivalent members. If the rewrite is a top-level rewrite then the lemma follows by lemmas 6.1, 7.4, 7.5, and 7.6. If however r is a rewrite of a strict subterm then the lemma follows easily by the induction hypothesis and lemma 7.2. Decidability follows as one can prove  $t =_c t'$  by enumerating and comparing the sets NF(t) and NF(t'), while given a span from t, a co-span can be constructed to any member of NF(t).

# 8. An Extension of $\beta$ -reduction

A rewrite relation consisting of  $\beta$ -reductions, commuting conversions and restricted  $\eta$ expansions is defined which, when taken together with the conversion relation, generates the same equational theory as the full expansionary rewrite relation. This extension of  $\beta$ -reduction is proved strongly normalising and confluent by generalising the proof in (C. B. Jay and N. Ghani 1995) for the fragment without coproducts.

Firstly define the rewrite relation  $\Rightarrow_{\beta}$  to be the least congruence containing the reductions of Table 6. The rewrite relation  $\Rightarrow_{\beta}$  is known to be strongly normalising and confluent (D. Prawitz 1971). As mentioned before, a limited form of  $\eta$ -expansion is defined via the following function on terms:

$$\eta(t) = \begin{cases} \langle \pi_0 t, \pi_1 t \rangle & \text{if } t \text{ is of product type} \\ \mathbf{case}(t, x.\mathbf{in}_1(x), y.\mathbf{in}_2(y)) & \text{if } t \text{ is of sum type} \\ * & \text{if } t \text{ is of unit type} \\ \lambda x.tx & \text{if } t \text{ is of function type} \end{cases}$$

The  $\eta$ -expansion of terms of sum type given above is a special case of the  $\eta_+$ -rewrite rule in Table 2 and converts a (sub)term of sum type into a conversion which may then be expanded by the conversion relation. Uncontrolled  $\eta$ -expansion is clearly not strongly normalising and so restrictions must be imposed on their scope and in particular the expansions appearing in the triangle laws must be prevented. For the sum type, these triangle laws assert that expanding an injection term or a negative occurrence, i.e. the first argument of a case-expression, forms a looping reduction. However these restrictions are insufficient to obtain a strongly normalising relation as the  $\eta$ -expansion of a caseexpression of sum type reduces to the term obtained by expanding the arms of the

| Table 7. The Restricted Rewrite Relation                                                      |                                                                                               |
|-----------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------|
| $t  \text{expandable}  t \Rightarrow_{\beta} t'$                                              | $t \Rightarrow_{\mathcal{I}} t'$                                                              |
| $t \Rightarrow_{\mathcal{F}} \eta(t) \qquad t \Rightarrow_{\mathcal{I}} t'$                   | $t \Rightarrow_{\mathcal{F}} t'$                                                              |
| $u \Rightarrow_{\mathcal{F}} u'$                                                              | $v \Rightarrow_{\mathcal{F}} v'$                                                              |
| $\overline{\langle u, v \rangle \Rightarrow_{\mathcal{I}} \langle u', v \rangle}$             | $\overline{\langle u, v \rangle \Rightarrow_{\mathcal{I}} \langle u, v' \rangle}$             |
| $t \Rightarrow_{\mathcal{I}} t'$                                                              | $t \Rightarrow_{\mathcal{I}} t'$                                                              |
| $\pi_0 t \Rightarrow_{\mathcal{I}} \pi_0 t'$                                                  | $\pi_1 t \Rightarrow_{\mathcal{I}} \pi_1 t'$                                                  |
| $t \Rightarrow_{\mathcal{I}} t'$                                                              | $u \Rightarrow_{\mathcal{F}} u'$                                                              |
| $\overline{tu \Rightarrow_{\mathcal{I}} t'u}$                                                 | $tu \Rightarrow_{\mathcal{I}} tu'$                                                            |
| $t \Rightarrow_{\mathcal{F}} t'$                                                              | $t \Rightarrow_{\mathcal{I}} t'$                                                              |
| $\overline{\lambda x.t \Rightarrow_{\mathcal{I}} \lambda x.t'}$                               | $\mathbf{case}(t, x.u, y.v) \Rightarrow_{\mathcal{I}} \mathbf{case}(t', x.u, y.v)$            |
| $u \Rightarrow_{\mathcal{F}} u'$                                                              | $v \Rightarrow_{\mathcal{F}} v'$                                                              |
| $\overline{\mathbf{case}(t, x.u, y.v)} \Rightarrow_{\mathcal{I}} \mathbf{case}(t, x.u', y.v)$ | $\overline{\mathbf{case}(t, x.u, y.v)} \Rightarrow_{\mathcal{I}} \mathbf{case}(t, x.u, y.v')$ |
| $t \Rightarrow_{\mathcal{F}} t'$                                                              | $t \Rightarrow_{\mathcal{F}} t'$                                                              |
| $\overline{\mathbf{in}_1(t) \Rightarrow_{\mathcal{I}} \mathbf{in}_1(t')}$                     | $\mathbf{in}_2(t) \Rightarrow_{\mathcal{I}} \mathbf{in}_2(t')$                                |

case-expression. If these arms are injections, a reduction loop may be formed, e.g.

$$\begin{aligned} \mathbf{case}(t, x.\mathbf{in}_1(x), y.\mathbf{in}_2(y)) & \Rightarrow_{\eta} \quad \eta(\mathbf{case}(t, x.\mathbf{in}_1(x), y.\mathbf{in}_2(y))) \\ & \Rightarrow_{\beta} \quad \mathbf{case}(t, x.\eta(\mathbf{in}_1(x)), y.\eta(\mathbf{in}_2(y))) \\ & \Rightarrow_{\beta}^* \quad \mathbf{case}(t, x.\mathbf{in}_1(x), y.\mathbf{in}_2(y)) \end{aligned} \tag{9}$$

Such terms are called *quasi-introduction* terms and the set of terms which may not be  $\eta$ -expanded is enlarged to include them. The *quasi-introduction* terms of sum type are defined by the syntax:

$$q := \mathbf{in}_1(t) \mid \mathbf{in}_2(t) \mid \mathbf{case}(t, x.q, y.q')$$

where t ranges over arbitrary terms and q, q' over arbitrary quasi-introduction terms of sum type. Given any quasi-introduction term of sum type, the introduction terms at its leafs are extracted by the function Arm which is defined as follows:

$$Arm(t) = \begin{cases} Arm(t_1) \cup Arm(t_2) & \text{if } t = \mathbf{case}(u, x.t_1, y.t_2) \\ \{t\} & \text{otherwise} \end{cases}$$

Because the  $\eta$ -expansion of terms such as  $case(u, x.\lambda x'.t_1, y.\lambda y'.t_2)$  do not create a reduction loop as in equation 9, such terms are regarded as expandable and so the quasiintroduction terms of product, exponential and unit type are defined to be just the introduction terms of that type.

A term is *expandable* providing it is neither a quasi-introduction term nor of base type. The reduction relation  $\Rightarrow_{\mathcal{F}}$  is defined simultaneously with a subrelation  $\Rightarrow_{\mathcal{I}}$  which is guaranteed not to be a top-level expansion — hence a negative occurrence may be  $\Rightarrow_{\mathcal{I}}$ -rewritten when a  $\Rightarrow_{\mathcal{F}}$ -reduction may create a reduction loop. The definitions of these relations are contained in Table 7. Note that if t is a quasi-introduction term, then so are its  $\Rightarrow_{\mathcal{F}}$ -reducts and  $\eta(t) \Rightarrow_{\beta}^{*} t$ . In addition,  $\Rightarrow_{\mathcal{F}}$ -normal forms satisfy the following

structural properties which make them the natural candidate for the generalisation of long  $\beta\eta$ -normal form to this calculus.

**Lemma 8.1.** The normal forms of  $\Rightarrow_{\mathcal{F}}$  are exactly those terms which are  $\beta$ -normal forms and each of whose subterms are either of base type, occur negatively, or a quasi-introduction term.

*Proof.* The proof is by induction on term structure.

The relations  $\Rightarrow_{\mathcal{I}}$  and  $\Rightarrow_{\mathcal{F}}$  are not congruences and this complicates normalisation and confluence proofs. The following lemma characterises how substitutivity may fail.

**Lemma 8.2.** Let t, t', u and u' be terms such that  $t \Rightarrow_{\mathcal{R}} t'$  and  $u \Rightarrow_{\mathcal{R}} u'$ , where  $\mathcal{R} \in \{\mathcal{I}, \mathcal{F}\}$ . Then

- There is a rewrite  $t[u/x] \Rightarrow_{\mathcal{R}} t'[u/x]$  unless u is a quasi-introduction term and t' is obtained by expanding an occurrence of x in t. In this case there are reduction sequences  $t[\eta(u)/x] \Rightarrow_{\mathcal{I}}^* t'[u/x] \Rightarrow_{\mathcal{I}}^* t[u/x]$ .
- There is a rewrite  $t[u/x] \Rightarrow_{\mathcal{I}}^* t[u'/x]$  unless  $u' = \eta(u)$  and either t = x or there are negative occurrences of x in t. In this latter case t[u'/x] and t[u/x] have a common  $\Rightarrow_{\mathcal{I}}^*$ -reduct.

*Proof.* Induction on the definition of the rewrites.

The obvious next step would be to hypothesise that both  $\Rightarrow_{\mathcal{I}}$  and  $\Rightarrow_{\mathcal{F}}$  are locally confluent. Unfortunately this is not the case:

$$\begin{array}{ccc} (\lambda x.t)u \xrightarrow{\mathcal{I}} & (\lambda x.\eta(t))u \\ \mathcal{I} & & \downarrow \mathcal{I} \\ t[u/x] \xrightarrow{\mathcal{F}} & \eta(t)[u/x] \end{array}$$

In these examples the bottom arrow is  $\Rightarrow_{\mathcal{F}}^*$ , but not  $\Rightarrow_{\mathcal{I}}^*$ , and so  $\Rightarrow_{\mathcal{I}}$  is not locally confluent. However local confluence of  $\Rightarrow_{\mathcal{F}}$  can be proved in conjunction with a slight variant for  $\Rightarrow_{\mathcal{I}}$ .

**Lemma 8.3.** The relation  $\Rightarrow_{\mathcal{F}}$  is locally confluent and given any divergence  $t \Rightarrow_{\mathcal{I}} t_i$  (where i = 1, 2), there is a term t' such that  $t_1 \Rightarrow_{\mathcal{I}}^* t'$  or  $t_1 \Rightarrow_{\mathcal{F}} t'$  and similarly, either  $t_2 \Rightarrow_{\mathcal{I}}^* t'$  or  $t_2 \Rightarrow_{\mathcal{F}} t'$ .

*Proof.* The proof is by simultaneous induction on the term t, with the tricky cases handled by lemma 8.2.

The substitutivity lemma 8.2 also allows us to give a simple definition of the  $\Rightarrow_{\mathcal{F}}^*$  reducts of a variable via a function  $\Delta$ , which maps variables to sets of terms:

| $\Delta(z) = \{z\}$                                                                                                                               | z has base type  |
|---------------------------------------------------------------------------------------------------------------------------------------------------|------------------|
| $\Delta(z) = \{z, *\}$                                                                                                                            | z has unit type  |
| $\Delta(z) = \{z\} \cup \{ \langle \alpha[\pi_0 z/x], \alpha'[\pi_1 z/y] \rangle \mid \alpha \in \Delta(x) \text{ and } \alpha' \in \Delta(y) \}$ | z has prod. type |
| $\Delta(z) = \{z\} \cup \{\lambda x.\alpha'[z\alpha/y] \mid \alpha \in \Delta(x) \text{ and } \alpha' \in \Delta(y)\}$                            | z has fun. type  |

and if z has sum type, then

$$\Delta(z) = \{z\} \cup \{\mathbf{case}(z, x.\mathbf{in}_1(\alpha), y.\mathbf{in}_2(\alpha')) \mid \alpha \in \Delta(x) \text{ and } \alpha' \in \Delta(y)\}$$

| Table 8. Definition of Validity                                                                                               |
|-------------------------------------------------------------------------------------------------------------------------------|
| — If $t$ is a variable, constant, application or projection                                                                   |
| $t/\Rightarrow_{\mathcal{I}} \subseteq V(X)$                                                                                  |
| $t \in V(X)$                                                                                                                  |
| — If t is a pair                                                                                                              |
| $u_i \in V(X_i)  \langle u_0, u_1 \rangle / \Rightarrow_{\mathcal{I}} \subseteq V(X_0 \times X_1)$                            |
| $\langle u_0, u_1 \rangle \in V(X_0 \times X_1)$                                                                              |
| - If t is a $\lambda$ -abstraction                                                                                            |
| $\forall u \in V(X) . t[u/x] \in V(Y)  \lambda x . t/ \Rightarrow_{\mathcal{I}} \subseteq V(X \to Y)$                         |
| $\lambda x.t \in V(X \to Y)$                                                                                                  |
| - If $t$ is an injection                                                                                                      |
| $t \in V(X_i)$ $\mathbf{in}_i(t) / \Rightarrow_{\mathcal{I}} \subseteq V(X_1 + X_2)$                                          |
| $\mathbf{in}_i \left( t \right) \in V(X_1 + X_2)$                                                                             |
| - If $t = case(t_0, x_1.u_1, x_2.u_2)$ is a case-expression of function type                                                  |
| $u_i \in V(X \rightarrow Y)$ if $t_0 = \mathbf{in}_i(v)$ then $u_i[\mathcal{I}v/x_i] \in V(T)$                                |
| $t/\Rightarrow_{\mathcal{I}}\subseteq V(X\rightarrow Y)$ case $(t_0, x_1.u_1s, x_2.u_2s)\in V(X_i)$                           |
| $\mathbf{case}(t_0, x_1.u_1, x_2.u_2) \in V(X \to Y)$                                                                         |
| - If $t = case(t_0, x_1.u_1, x_2.u_2)$ is a case-expression of product type                                                   |
| $u_i \in V(X \times Y)$ if $t_0 = \mathbf{in}_i(v)$ then $u_i[\mathcal{I}v/x_i] \in V(T)$                                     |
| $t/\Rightarrow_{\mathcal{I}}\subseteq V(X\times Y)$ $\mathbf{case}(t_0, x_1.\pi_i u_1, x_2.\pi_i u_2)\in V(X_i)$              |
| $\mathbf{case}(t_0, x.u, y.v) \in V(X_0 \times X_1)$                                                                          |
| - If $t = case(t_0, x_1.u_1, x_2.u_2)$ is a case-expression not of function or product type                                   |
| $u_i \in V(X)$ $t/\Rightarrow_{\mathcal{I}} \subseteq V(X)$ if $t_0 = \mathbf{in}_i(v)$ then $u_i[\mathcal{I}v/x_i] \in V(T)$ |
| $case(t_0, x_1.u_1, x_2.u_2) \in V(X)$                                                                                        |

where the variables x and y have the appropriate type. The function  $\Delta$  is extended to terms by:

$$\Delta(t) = \{t_0[t/z] \mid t_0 \in \Delta(z)\}$$

**Lemma 8.4.** The function  $\Delta$  gives the rewrites of a variable, i.e.  $\Delta(z) = z/\Rightarrow_{\mathcal{F}}^*$ .

*Proof.* The proof is by induction on the type of z.

The relation  $\Rightarrow_{\mathcal{F}}$  is proved strongly normalising by extending the proof of strong normalisation for fragment without coproducts presented in (C. B. Jay and N. Ghani 1995). The set of *valid* terms of type T is denoted V(T) and defined by induction over T. For each type T, V(T) consists of those terms which can be shown valid by the inference rules in Table 8, where t/R is the set of one-step R-reducts of t. Notice that in order to prove  $\operatorname{case}(\operatorname{in}_1(t), x.u, y.v)$  valid, we must not only prove that the reduct u[t/x] is valid, but also that  $u[\mathcal{I}t/x]$  is valid, where  $\mathcal{I}$  is the operator on terms defined by

$$\mathcal{I}(t) = (\lambda x.x)t$$

This is because a reduction  $u \Rightarrow_{\mathcal{F}} u'$  will not always induce one  $u[t/x] \Rightarrow_{\mathcal{F}} u'[t/x]$  but, because  $\mathcal{I}(t)$  is a neutral term, there will always be a rewrite  $u[\mathcal{I}t/x] \Rightarrow_{\mathcal{F}} u'[\mathcal{I}t/x]$ . In

particular this will be needed to prove lemma 8.7. Notice also that as variables have no  $\Rightarrow_{\mathcal{I}}$ -reducts, the valid terms of a given type contain all the variables of that type.

The valid terms of a given type are shown to satisfy the following validity predicates defined over sets of terms P:

- V1: If  $t \in P$  then t is  $\Rightarrow_{\mathcal{F}}$ -strongly normalising
- V2: If  $t \in P$  and  $t \Rightarrow_{\mathcal{F}} t'$  then t' is valid
- V3: If  $t \in P$  then  $\Delta(t) \subseteq P$
- V4: If  $t \in P$  then  $\mathcal{I}(t) \in P$

Although the predicate V4 has been included as a separate predicate, if P is the set of valid terms of some type, then V4 is actually a consequence of the first three validity predicates.

**Lemma 8.5.** Assume the valid terms of a given type satisfy the validity predicates V1-3 and let t be a valid term of that type. Then the term  $\mathcal{I}(t)$  is also valid.

*Proof.* We prove the stronger assertion that if x is any variable having the same type as t, then for any  $\alpha \in \Delta(x)$  the term  $(\lambda x.\alpha)(t)$  is valid. By V3 the term  $\alpha$  is valid and so by V1 the terms  $\alpha$  and t are strongly normalising and hence the sum of their normalisation ranks may be used as an induction rank. The  $\Rightarrow_{\mathcal{I}}$ -reducts of  $(\lambda x.\alpha)(t)$  induced by reductions of  $\alpha$  or t are valid by the induction hypothesis while the only other  $\Rightarrow_{\mathcal{I}}$ -reduct is  $\alpha[t/x]$  which is a member of  $\Delta(t)$  and so valid by V3.

We now establish the validity predicates for terms of sum type.

**Lemma 8.6.** Assume V(X) and V(Y) satisfy V1, V2 and V3. Then the set of valid quasi-introduction terms of type X + Y satisfies V1 and V2. Also if u: X and v: Y are valid, then so are  $in_1(u)$  and  $in_2(v)$ .

*Proof.* Since quasi-introduction terms are non-expandable and closed under reduction, the first part of the lemma follows by induction on validity. The second half follows by induction on the normalisation ranks of u and v.

**Lemma 8.7.** Assume V(X) and V(Y) satisfy the validity predicates V1-3 and let t be a valid term of type X + Y.

- If  $t = \operatorname{case}(t_0, x.u, y.v), \eta(u) \Rightarrow_{\mathcal{F}}^* \alpha \text{ and } \eta(v) \Rightarrow_{\mathcal{F}}^* \beta \text{ then } \operatorname{case}(t_0, x.\alpha, y.\beta) \text{ is valid.}$ - All terms  $t' \in \Delta(t)$  are valid.

*Proof.* The proof is by simultaneous induction on the validity of t.

(i) By the induction hypothesis the terms η(u) and η(v) are both valid, quasi-introduction terms and hence so are α and β. Thus the sum of their normalisation ranks forms an inner induction rank. Those ⇒<sub>I</sub>-reducts of case(t<sub>0</sub>, x.α, y.β) induced by reductions of proper subterms are valid by the induction hypothesis and this leaves two cases. If t<sub>0</sub> is an introduction term, and say case(in<sub>1</sub>(s), x.α, y.β) ⇒<sub>I</sub> α[s/x] then note first that u[Is/x] is valid. By induction so is η(u[Is/x]) = η(u)[Is/x] and, as α[Is/x] is a reduct of this term, it is also valid. Hence α[s/x] is valid. Similarly if t<sub>0</sub> is a case-expression, the result of a commuting conversion is shown valid by applying the induction hypothesis to the term obtained by contracting the top level commuting conversion in t.

# $\beta\eta$ -Equality for Coproducts

(ii) We must prove that  $\operatorname{case}(t, x \cdot \operatorname{in}_1(u), y \cdot \operatorname{in}_2(v))$  is valid where  $u \in \Delta(x)$  and  $v \in \Delta(y)$ . By the induction hypothesis u and v are valid and strongly normalising and hence the sum of their normalisation ranks forms an inner induction rank. The validity of  $\operatorname{in}_1(u)$  and  $\operatorname{in}_2(v)$  follow by lemma 8.6 from the validity of u and v while those  $\Rightarrow_{\mathcal{I}}$ -reducts induced by reductions of proper subterms are valid by the induction hypothesis. The result of a top-level commuting conversion is valid by the first half of this lemma and, finally, if t is an injection, say  $\operatorname{in}_1(t_0)$ , then  $t_0$  is a valid term of type X and hence by lemma 8.5 so is  $\mathcal{I}t_0$ . By V3  $u[\mathcal{I}t_0/x]$  is also valid and hence by lemma 8.6  $\operatorname{in}_1(u)[\mathcal{I}t_0/x]$  is valid.

**Corollary 8.8.** Assume V(X) and V(Y) satisfy the validity predicates V1-3. Then so do the valid terms of type X + Y.

*Proof.* Let t be a term. The lemma is established by induction on the validity of t. All  $\Rightarrow_{\mathcal{I}}$ -reducts are valid and, by the induction hypothesis, strongly normalising. The only other reduct is a valid quasi-introduction term which is strongly normalising by lemma 8.6. Thus all reducts of t are strongly normalising and hence so is t. The  $\Rightarrow_{\mathcal{I}}$ -reducts of a term are valid by definition, while the result of a basic expansion is valid from lemma 8.7. Finally, V3 has just been established in lemma 8.7.

**Lemma 8.9.** The set of valid terms of every type satisfy the three validity predicates V1, V2 and V3.

*Proof.* The proof is by induction on the type of a term. Terms of base type are proved strongly normalising by induction on their validity and because such terms have no expansions the predicates V2 and V3 are also satisfied. Similar remarks apply to terms of unit type as their only expansion is the valid constant \* which is also a normal form and so strongly normalising. Terms of sum type have just been shown to satisfy the validity predicates while the arguments for terms of function and product type are similar to those of (C. B. Jay and N. Ghani 1995).

Before showing all terms are valid, the criteria for proving a *case*-expression valid is simplified.

**Lemma 8.10.** The term  $\operatorname{case}(t, x_1.u_1, x_2.u_2)$  is valid iff t is strongly normalising,  $u_1, u_2$  are valid, and in addition if  $t \Rightarrow_{\mathcal{I}}^* t'$  and  $\operatorname{in}_i \alpha \in \operatorname{Arm}(t')$  then  $u_i[\mathcal{I}\alpha/x_i]$  is valid.

*Proof.* The proof is by induction with rank the quadruple of numbers (a, b, c, d), where a is the complexity of the type of the case-expression, b is the  $\Rightarrow_{\mathcal{F}}$ -normalisation rank of t, c is the size of t and d is the sum of the  $\Rightarrow_{\mathcal{F}}$ -normalisation ranks of  $u_1$  and  $u_2$ . There are two proof obligations. Firstly if the case-expression is of sum or function type, the clauses pertaining to commuting conversions are easily established by the induction hypothesis. Secondly, those  $\Rightarrow_{\mathcal{I}}$ -reducts induced by reductions of proper subterms are valid by the induction hypothesis with the second part of the induction hypothesis following from lemma 8.2, while a basic  $\beta$ -reduction has a valid reduct by assumption. Finally, if t is a case-expression then the result of a basic commuting conversion is shown valid by first using the induction hypothesis to prove the arms valid and then once more for the whole term.

Finally, all terms are shown valid in the traditional manner:

**Lemma 8.11.** Let t be a term and  $u_i$  be valid terms. Then the term  $t[u_i/x_i]$  is a valid term.

*Proof.* The proof is by induction on t and follows the standard pattern. The only interesting part is for the term case(t', x.u, y.v). The terms  $u[u_i/x_i]$ ,  $v[u_i/x_i]$  and  $t'[u_i/x_i]$  are valid and thus strongly normalising by the induction hypothesis. Thus if  $t'[u_i/x_i] \Rightarrow_{\mathcal{I}}^{*}$  t'' and  $in_1(\alpha) \in Arm(t'')$ , we may deduce  $\mathcal{I}\alpha$  is valid and thus by the induction hypothesis

$$(u[u_i/x_i])[\mathcal{I}\alpha/x] = u[u_i/x_i, \mathcal{I}\alpha/x]$$

is a valid term. Similar considerations apply to right injections and so the lemma is proven.  $\hfill \square$ 

**Theorem 8.12.** The relations  $\Rightarrow_{\mathcal{F}}$  and  $\Rightarrow_{\mathcal{I}}$  are strongly normalising and  $\Rightarrow_{\mathcal{F}}$  is confluent.

*Proof.* As variables are valid, all terms are proven valid by instantiating lemma 8.11 with the identity substitution. Hence all terms are  $\Rightarrow_{\mathcal{F}}$ -strongly normalising. Confluence now follows from local confluence proved in lemma 8.3 and strong normalisation. of  $\Rightarrow_{\mathcal{F}}$ .

# 9. Decidability of $\beta\eta$ -Equality

The expansionary rewrite relation defined in Table 2 has been decomposed into the strongly normalising and confluent relation  $\Rightarrow_{\mathcal{F}}$  and the decidable conversion relation. The rest of this paper proves that the expansionary rewrite relation is itself decidable by showing that if two terms are  $\beta\eta$ -equivalent, then their  $\Rightarrow_{\mathcal{F}}$ -normal forms are equivalent in the conversion relation.

The easiest proof strategy would be to consider the  $\beta$ -reductions of a term in isolation from the possibilities for  $\eta$ -expansion that exist within the term, i.e. prove that  $\eta$ -expansion preserves  $\beta$ -normal forms and that if  $t =_c t'$ , then  $\beta(t) =_c \beta(t')$  and  $\eta(t) =_c \eta(t')$ , where  $\beta(t)$  denotes the  $\beta$ -normal form of t and  $\eta(t)$  denotes the  $\eta$ -normal form of t. Unfortunately, unlike the restricted expansions of the calculus without coproducts, the  $\eta$ -expansions contained in  $\Rightarrow_{\mathcal{F}}$  do not form a confluent relation, e.g. the reducts of the span below cannot be rewritten to the same term using restricted expansions alone.

$$\eta(\mathbf{case}(z, x.x, y.y)) \Leftarrow \mathbf{case}(z, x.x, y.y) \Rightarrow^* \mathbf{case}(z, x.\eta(x), y.\eta(y))$$

Our solution is to define a function which picks a particular  $\eta$ -normal form for a term by preventing the  $\eta$ -expansion of <u>all</u> case-expressions. The conversion relation may also map positive occurrences to negative occurrences, e.g. in the following rewrite the subterms x and y occur positively in the redex but negatively in the reduct:

$$(\mathbf{case}(z, x.x, y.y))w \Rightarrow_c \mathbf{case}(z, x.xw, y.yw)$$

Such reductions cannot be lifted to their  $\eta$ -normal forms, i.e. there is no reduction

 $(\mathbf{case}(z, x.\eta(x), y.\eta(y)))w /\Rightarrow_c \mathbf{case}(z, x.xw, y.yw)$ 

The solution to this second problem is to increase those occurrences in a term which may be  $\Rightarrow_{\mathcal{I}}$ -rewritten but not  $\Rightarrow_{\mathcal{F}}$ -rewritten. Rather than present another series of relations, we define the fully  $\eta$ -expanded form of a term directly via the simultaneous definition of a pair of functions  $\eta \mathcal{F}$  and  $\eta \mathcal{I}$ .

$$\begin{split} \eta \mathcal{I}(x) &= x \\ \eta \mathcal{I}(*) &= * \\ \eta \mathcal{I}(uv) &= \eta \mathcal{I}(u) \eta \mathcal{F}(v) \\ \eta \mathcal{I}(\lambda x.t) &= \lambda x.\eta \mathcal{F}(t) \\ \eta \mathcal{I}(\pi_i t) &= \pi_i \eta \mathcal{I}(t) \\ \eta \mathcal{I}(\langle u, v \rangle) &= \langle \eta \mathcal{F}(u), \eta \mathcal{F}(v) \rangle \\ \eta \mathcal{I}(\mathbf{in}_i(t)) &= \mathbf{in}_i(\eta \mathcal{F}(t)) \\ \eta \mathcal{I}(\mathbf{case}(t, x.u, y.v)) &= \mathbf{case}(\eta \mathcal{I}(t), x.\eta \mathcal{I}(u), y.\eta \mathcal{I}(v)) \end{split}$$

 $\operatorname{and}$ 

$$\eta \mathcal{F}(t) = \begin{cases} \mathbf{case}(\eta \mathcal{I}(t'), x.\eta \mathcal{F}(u), y.\eta \mathcal{F}(v)) & t \text{ is } \mathbf{case}(t', x.u, y.v) \\ \Delta^m(z)[\eta \mathcal{I}(t)/z] & t \text{ is a projection, application, variable} \\ t & \text{otherwise} \end{cases}$$

where z is any variable having the same type as t and  $\Delta^m(z)$  is the  $\Rightarrow_{\mathcal{F}}$ -normal form of z (the superscript m is to distinguish the term  $\Delta^m(z)$  from the set of terms  $\Delta(z)$ ). To maintain the strength on the equational theory, the  $\eta$ -expansion of *case*-expressions is simulated by a rewrite relation  $\Rightarrow_{\delta}$  which is defined to be the least congruence containing the reductions

$$\begin{aligned} & \mathbf{case}(t, x.\lambda x'.u, y.\lambda x'.v) \quad \Rightarrow_{\delta} \quad \lambda x'.\mathbf{case}(t, x.u, y.v) \\ & \mathbf{case}(t, x.\langle u_1, v_1 \rangle, y.\langle u_2, v_2 \rangle) \quad \Rightarrow_{\delta} \quad \langle \mathbf{case}(t, x.u_1, y.u_2), \mathbf{case}(t, x.v_1, y.v_2) \rangle \end{aligned}$$

The rewrite relation  $\Rightarrow_{\delta}$  is strongly normalising and confluent and so has unique normal forms. In addition, if two terms are  $\Rightarrow_{\delta}$ -equivalent, then they are also equivalent in the conversion relation because  $\Rightarrow_{\delta}^{-1} \subseteq \Rightarrow_{c}$ . We now give an algorithm for the calculation of  $\Rightarrow_{\mathcal{F}}$ -normal forms.

**Lemma 9.1.** The  $\Rightarrow_{\mathcal{F}}$ -normal form of a term t may be calculated by: (i) calculating the  $\beta$ -normal form of t; (ii) applying the function  $\eta \mathcal{F}$ ; and (iii) calculating the  $\Rightarrow_{\delta}$ -normal form of the result.

*Proof.* Let  $\sharp t$  denote the result of applying the algorithm in the lemma to t. The lemma is proved by showing that  $\sharp t$  is an  $\Rightarrow_{\mathcal{F}}$ -normal form and is also  $\Rightarrow_{\mathcal{F}}$ -equivalent to t. Firstly  $\sharp t$  is a  $\beta$ -normal form as both  $\eta \mathcal{F}$  and  $\Rightarrow_{\delta}$  preserve  $\beta$ -normal forms, while induction on the structure of t is used to show that if t is a  $\beta$ -normal form, then  $\sharp t$  is an  $\Rightarrow_{\mathcal{F}}$ -normal form. Secondly, because there is always a reduction sequence  $t \Rightarrow_{\mathcal{F}} \eta \mathcal{F}(t)$  and a term is always  $\Rightarrow_{\mathcal{F}}$ -equivalent to its  $\Rightarrow_{\delta}$ -reducts, we may conclude that  $t =_{\mathcal{F}} \sharp t$ .

We define a relation  $\rightarrow$  by parallelising the conversion relation and prove that if  $t \rightarrow t'$  then  $\beta(t) \rightarrow \beta(t')$ . In order to prove this, the parallelised conversion relation must

| Ta       | ble 9. The Parallel Conversion Relation                                                                                       |
|----------|-------------------------------------------------------------------------------------------------------------------------------|
| <b>—</b> | Identity                                                                                                                      |
|          | $\overline{z \twoheadrightarrow z}$                                                                                           |
| ⊢        | Expansion                                                                                                                     |
|          | $X \subseteq FC(t)$ is consistent $t/X \twoheadrightarrow u = t \setminus_i X \twoheadrightarrow v_i$                         |
|          | $t \twoheadrightarrow \mathbf{case}(u, x.v_1, y.v_2)$                                                                         |
|          | where $x, y$ are the variables bound by X. If however X is empty, then                                                        |
|          | $x, y \notin FV(t)$ and u is any term of the appropriate type.                                                                |
|          | A congruence rule for each term constructor                                                                                   |
|          | $u_0 \twoheadrightarrow u'_0, \ldots, u_n \twoheadrightarrow u'_n$                                                            |
|          | $\mathcal{T}(u_0,\ldots,u_n) \twoheadrightarrow \mathcal{T}(u'_0,\ldots,u'_n)$                                                |
| _        | A parallel conversion $t \twoheadrightarrow t'$ is said to be <i>full</i> , and written $t \twoheadrightarrow_f t'$ , iff the |
|          | left branch of any expansion or elimination congruence does not                                                               |
|          | itself end in an expansion.                                                                                                   |

permit the expansion of empty sets of conversions and so a parallelised version of the weakening clause is not needed. Parallel conversion, denoted  $\rightarrow$ , is defined to be the least congruence defined by the inference rules in Table 9. The full parallel conversion relation, denoted  $\rightarrow_f$ , is the subrelation of  $\rightarrow$  obtaining by insisting that the left-branch of any instance of expansion or a elimination congruence must not itself be an expansion. These restrictions ensure that  $\rightarrow_f$  does not introduce new commuting conversions and hence  $\rightarrow_f$  will preserve  $\beta$ -normal forms.

**Lemma 9.2.** Parallel expansion is closed under substitution, i.e. if  $t \to t'$  and  $u \to u'$  then  $t[u/x] \to t'[u'/x]$ . In addition if t is a  $\beta$ -normal form and  $t \to_f t'$  then t' is also a  $\beta$ -normal form. Finally if  $t \to t'$ , there is a term t'' such that  $t \to_f t''$  and  $t' \Rightarrow_{\beta}^{*} t''$ .

*Proof.* The proofs are all by induction on the term t.

**Lemma 9.3.** Let  $t \rightarrow t'$ . Then  $\beta(t) \rightarrow \beta(t')$ .

*Proof.* We prove that if  $t \to_f t'$  then  $\beta(t) \to_f \beta(t')$  by induction on firstly the  $\beta$ -normalisation rank of t and secondly the depth of the rewrite. The lemma then follows from lemma 9.2 since  $\rightarrow$  can be embedded in  $\rightarrow_f$ .

If t is a  $\beta$ -normal form then by lemma 9.2 t' is also a  $\beta$ -normal form. For the inductive step consider a parallel conversion of the form:

$$\frac{X \subseteq FC(t) \text{ is consistent } t/X \twoheadrightarrow_f u \quad t \setminus_i X \twoheadrightarrow_f v_i}{t \twoheadrightarrow_f \mathbf{case}(u, x.v_1, y.v_2)}$$

If X is empty, then by the induction hypothesis there are rewrites  $\beta(t) \rightarrow_f \beta(v_i)$  and hence a parallel conversion

$$\frac{\beta(t) \twoheadrightarrow_f \beta(v_i)}{\beta(t) \twoheadrightarrow \mathbf{case}(u, x.\beta(v_1), y.\beta(v_2))}$$

where we rely on  $\beta$ -reduction not to introduce new free variables. If however X is nonempty there are four subcases. Firstly if t/X is not a  $\beta$ -normal form, then by the induction

hypothesis there is a parallel rewrite  $\beta(t/X) \twoheadrightarrow_f \beta(u)$  and hence a rewrite

$$\frac{\beta(t/X) \twoheadrightarrow_f \beta(u) \quad t \setminus_i X \twoheadrightarrow_f v_i}{t[X \leftarrow \beta(t/X)] \twoheadrightarrow \mathbf{case}(\beta(u), x.v_1, y.v_2)}$$

which embeds to a full parallel conversion rewrite and the lemma then follows by the induction hypothesis. If however t/X is a  $\beta$ -normal form then consider the case where t/X is an injection, say  $\mathbf{in}_1(r)$ . Then by fullness, u must also be an injection, say  $\mathbf{in}_1(r')$  where  $r \rightarrow_f r'$ . Now by lemma 5.1, the result of contracting the  $\beta_+$ -redexes in t associated to X is  $(t \setminus_1 X)[r/x]$  and thus there are reductions

$$t \xrightarrow{f} \operatorname{case}(\operatorname{in}_{1}(r'), x.v_{1}, y.v_{2})$$

$$\beta^{*} \downarrow \qquad \beta^{*} \downarrow$$

$$(t \setminus_{1} X)[r/x] \xrightarrow{} v_{1}[r'/x]$$

where the bottom rewrite follows as parallel rewriting is closed under substitution. This rewrite can then be extended to a full rewrite to which the induction hypothesis may be applied. A third possibility is that t/X is a case expression, in which case a similar argument works — namely carry out the commuting conversions in t and at the top level of t' and then apply the induction hypothesis. Finally, if none of these cases are applicable, then an inductive argument proves there is free consistent set of conversions  $\beta(X) \subseteq FC(\beta(t))$  such that

$$\beta(t)/\beta(X) = \beta(t/X) = t/X$$
 and  $\beta(t) \setminus_i \beta(X) = \beta(t \setminus_i X)$ 

and thus there is a rewrite

$$\frac{\beta(t)/\beta(X) = \beta(t/X) \twoheadrightarrow \beta(u) \quad \beta(t) \setminus_i \beta(X) = \beta(t \setminus_i X) \twoheadrightarrow \beta(v_i)}{\beta(t) \twoheadrightarrow \mathbf{case}(\beta(u), x.\beta(v_1), y.\beta(v_2))}$$

which extends to a full parallel conversion rewrite which proves the lemma. Note that if  $\beta(X)$  is empty the proof is still valid as one of the conditions on the variables x, y bound by conversions  $X \subseteq FC(t)$  is that  $x, y \notin FV(t)$  and so  $x, y \notin FV(\beta(t))$ . If however the parallel rewrite has as last rule a congruence,

$$\frac{t_0 \twoheadrightarrow_f t'_0, \dots, t_n \twoheadrightarrow_f t'_n}{\mathcal{T}(t_0, \dots, t_n) \twoheadrightarrow_f \mathcal{T}(t'_0, \dots, t'_n)}$$

then there are three possibilities. If there is an immediate subterm which is not a  $\beta$ -normal form, then the induction hypothesis may be used on each of the subterms so that  $\mathcal{T}(\beta(t_0), \ldots, \beta(t_n)) \twoheadrightarrow \mathcal{T}(\beta(t'_0), \ldots, \beta(t'_n))$  and then the induction hypothesis invoked again. On the other hand if the only redex is a top level redex, then by fullness there is a also a redex at the top level of t'. There is a parallel rewrite between the terms obtained by performing these reductions and the lemma then follows by the induction hypothesis.

The second part of the embedding theorem concerns the interaction between the conversion relation and the  $\eta$ -expansions implicit in the function  $\eta \mathcal{F}$ . The key lemma is the following:

**Lemma 9.4.** Given a conversion  $\sigma \in FC(t)$  there is a consistent set of free conversions  $\sigma^{\mathcal{R}} \subseteq FC(\eta \mathcal{R}(t))$  such that

$$\eta \mathcal{R}(t) / \sigma^{\mathcal{R}} = \eta \mathcal{I}(t/\sigma) \text{ and } \eta \mathcal{R}(t) \setminus_{i} \sigma^{\mathcal{R}} = \eta \mathcal{R}(t \setminus_{i} \sigma)$$

where  $\mathcal{R} \in \{\mathcal{I}, \mathcal{F}\}$ .

*Proof.* The lemma is proved by induction on the definition of the functions  $\eta \mathcal{F}$  and  $\eta \mathcal{I}$ .

Note that the  $\eta$ -rule for products duplicates its argument and hence the requirement that  $\sigma^{\mathcal{R}}$  be a set of conversions.

**Lemma 9.5.** Assume  $t \twoheadrightarrow t'$ . Then  $\eta \mathcal{R}(t) \twoheadrightarrow \eta \mathcal{R}(t')$  where  $\mathcal{R} \in \{\mathcal{I}, \mathcal{F}\}$ .

*Proof.* The lemma is proved simultaneously by induction on the rewrite  $t \rightarrow t'$ .

**Theorem 9.6.** If two terms are equivalent in the conversion relation, then so are their  $\Rightarrow_{\delta}$  normal forms. Thus the expansionary rewrite relation is decidable and confluent.

*Proof.* The first half of the lemma is trivial as  $\Rightarrow_{\delta}$  is contained in the inverse of the conversion relation.

Thus terms equivalent in the equational theory have  $\Rightarrow_{\mathcal{F}}$ -normal forms which are equivalent in the conversion relation which has already been shown to be decidable and confluent. Thus the expansionary rewrite relation is confluent and  $\beta\eta$ -equality is decidable.

#### 10. Conclusions and Further Work

In this paper an extensional equality for terms of **ABCC** was given. Each term has a finite set of quasi-normal reducts which are computable in two stages; firstly by  $\beta\eta$ normalisation and secondly by expanding as many conversions as possible. As terms equivalent in the equational theory have the same set of quasi-normal reducts, comparison of these normal forms provides a decision procedure for equality of terms.

There are several directions in which this research may be extended. Firstly the inability to define a unique normal form is closely linked to form of the *case*-expression which permits the elimination of one term at a time. An alternate, parallel elimination, allowing the concurrent elimination of several terms should permit the definition of unique normal forms and this is the subject of current work.

In a different direction, expansionary  $\eta$ -rewrite rules have already been applied to the more expressive members of the  $\lambda$ -cube (N. Ghani 1995a; N. Ghani 1996) and current research focuses on the addition of algebraic rewrite systems to these theories and also the more general Pure Type Systems. The techniques developed in this paper also seem to be applicable to the bang ! operator from linear logic although research here is at a preliminary stage.

### 11. Acknowledgements

I would like to thank my first supervisor Barry Jay who first introduced me to the applications of category theory in term rewriting. Roberto Di Cosmo has provided many

insights and suggestions which have been of enormous benefit to this work. I would also like to thank Stefan Kahrs, Don Sanella, and Christoph Luth for their considerable help and support on many occasions. Lastly Kevin Keegan has also helped in many ways.

#### References

- Y. Akama (1993), "On Mints' Reduction for ccc-Calculus". In "Typed λ-Calculus and Applications", LNCS 664, pages 1-12. Springer Verlag.
- H. Barendregt (1984), "The Lambda Calculus: Its Syntax and Semantics (revised edition)", Studies in Logic and the Foundations of Mathematics number 103. North Holland.
- N. Dershowitz and J.P. Jouannaud (199), "Rewrite Systems". In "The Handbook of Theoretical Computer Science, Vol. B: Formal Models and Semantics", chapter 6, pages 243-320. MIT Press.
- R. Di Cosmo and D. Kesner (1994), "Combining Algebraic Rewriting, Extensional Lambda Calculi and Fixpoints". To appear in Theoretical Computer Science. Also in ICALP LNCS 820, pages 462-472. Springer-Verlag.
- R. Di Cosmo and D. Kesner (1994), "Simulating Expansions Without Expansions". In Mathematical Structures in Computer Science, volume 4, pages 1-48.
- D. Dougherty (1993), "Some  $\lambda$ -Calculi with Categorical Sums and Products". In "Rewriting Techniques and Applications", LNCS 690 pages 137-151. Springer-Verlag.
- D. Dougherty and R. Subrahmanyam (1995), "Equality Between Functionals in the Presence of Coproducts". To appear in LICS.
- H. Friedman (1975), "Equality Between Functionals". In "Logic Colloquium", Springer-Verlag.
- N. Ghani (1995a), "Eta-Expansions III  $F^{\omega}$  ", Submitted to CSL'96.
- N. Ghani (1995b), "Adjoint Rewriting" Ph. D. Thesis, University of Edinburgh. Awaiting Publication.
- N. Ghani (1996), "Eta-Expansions IV The Calculus of Constructions", Unpublished.
- J. Y. Girard and P. Taylor and Y. Lafont (1989), "Proofs and Types". Cambridge Tracts in Theoretical Computer Science, volume 7. CUP.
- J. W. Gray (1974), "Formal Category Theory: Adjointness for 2-Categories". In Lecture Notes in Mathematics volume 391. Springer-Verlag.
- J. R. Hindley and J. P. Seldin (1986), "Introduction to Combinators and  $\lambda$ -Calculus". LMS Student Tracts, number 1. CUP.
- G. Huet (1976), "Résolution d'équations dans des langages d'ordre  $1, 2, ..., \omega$ ". Thèse d'Etat, Université de Paris VII.
- G. Huet (1980),"Confluent Reductions: Abstract Properties and Applications to Term Rewriting Systems". In JACM, volume 27, pages 797-821.
- C. B. Jay (1988), "Local Adjunctions". In JPAA volume 53, pages 227-238.
- C. B. Jay (1992), "Modelling Reduction in Confluent Categories". In "Applications of Categories in Computer Science", LMS Lecture Note Series, volume 177.
- C. B. Jay and N. Ghani (1995), "The Virtues of Eta-expansion". In "Journal of Functional Programming", volume 5(2), April 1995, pages 135-154. CUP.
- J. Lambek and P. J. Scott (1986), "Introduction to Higher Order Categorical Logic". Cambridge Studies in Advanced Mathematics, volume 7. CUP.
- G. E. Mints (1979), "Teorija Categorii i Teoria Dokazatelstv.I". In "Aktualnye problemy logiki i metodologii nauky", pages 252-278.
- D. Prawitz (1971), "Ideas and Results in Proof Theory". In "Proc. 2nd Scandinavian Logic Symposium", pages 235-307. North Holland.

- D. E. Rydeheard and J. G. Stell (1987), "Foundations of Equational Deduction: A Categorical Treatment of Equational Proofs and Unification Algorithms". In "Category Theory and Computer Science", LNCS 283, pages 114-139. Springer-Verlag.
- R. A. G. Seely (1987), "Modelling Computations: A 2-Categorical Framework". In "Proc. 2nd Annual Symposium on Logic in Computer Science", IEEE publications.
- V. Breazu-Tannen (1988), "Combining Algebra and Higher-Order Types". In LICS 1988, pages 82-90.
- V. Breazu-Tannen (1994), "Polymorphic Rewriting Preserves Algebraic Confluence". In Information and Computation 1994.