

Eta Expansions in System F

Neil Ghani

LIENS-DMI, Ecole Normale Supérieure
45, Rue D'Ulm, 75230 Paris Cedex 05, France
e-mail:ghani@dmi.ens.fr

Résumé

Au cours des dernières années, l'utilisation de règles de réduction expansives pour η dans le cadre de différents lambda-calculs typés est devenue de plus en plus fréquente, au fur et à mesure que les avantages des expansions sur l'approche contractif sont devenus évidents. À part la décidabilité de l'égalité, les règles expansives donne une explication naturelle de la forme normale $\beta\eta$ -longue, se généralisent plus facilement aux autres constructeurs de types, préservent des propriétés importantes en combinaison avec d'autres systèmes de réécriture et ont une explication catégorique.

Ce travail étend au système F les résultats obtenus pour le lambda calcul typé simple λ -calculus, en prouvant normalisation forte et confluence d'une notion de réduction qui utilise les règles β traditionnelles avec des expansions η restreintes. En ce faisant, nous obtenons la forme normale $\beta\eta$ -longue exactement comme forme normale dans le système avec expansions restreintes.

Ces résultats sont une première étape vers la preuve du fait que les expansions sont compatibles avec des systèmes plus puissants dans le λ -cube et aussi qu'elles permettent une combinaison satisfaisante avec des systèmes qui contiennent à la fois $\beta\eta$ et de la réécriture algébrique.

Abstract

The use of expansionary η -rewrite rules in various typed λ -calculi has become increasingly common in recent years as their advantages over contractive η -rewrite rules have become apparent. Not only does one obtain the decidability of $\beta\eta$ -equality, but rewrite relations based on expansions give a natural interpretation of long $\beta\eta$ -normal forms, generalise more easily to other type constructors, retain key properties when combined with other rewrite relations, and are supported by a categorical theory of reduction.

This paper extends the initial results concerning the simply typed λ -calculus to System F, that is, we prove strong normalisation and confluence for a rewrite relation consisting of traditional β -reductions and η -expansions satisfying certain restrictions. Further, we characterise the second order long $\beta\eta$ -normal forms as precisely the normal forms of the restricted rewrite relation.

These results are an important step towards showing that η -expansions are compatible with the more powerful members of the λ -cube and also towards a smooth combination of type theories with $\beta\eta$ -equality and algebraic rewrite systems.

1 Introduction

Extensional equality for terms of the simply typed λ -calculus requires η -conversion, whose interpretation as a rewrite rule has traditionally been as a contraction:

$$\lambda x^T . f x \Rightarrow f \quad (x \notin \mathbf{FV}(f)) \quad (1)$$

When combined with the usual β -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 η -contractions behave badly when combined with rewrite rules arising from either algebraic rewrite systems or from other type constructors. For instance, the presence of the unit type with η -rewrite rule $t \Rightarrow *$ leads to a loss of confluence [12]. Specifically if f is a variable of type $1 \rightarrow 1$ then the following divergence cannot be completed.

$$\lambda x^1 . * \Leftarrow \lambda x^1 . f x \Rightarrow f \quad (2)$$

The combination of type theories with algebraic rewrite relations is another area where the use of η -contractions creates problems. For instance, while it is known that the combination of a confluent first order rewrite system and a type theory (e.g. the simply typed λ -calculus, System F) equipped with β -reduction is confluent [10], these results cannot be generalised to type theories equipped with η -contraction as confluence is invariably lost. For example, consider a single sort 1 with constants $f : 1 \rightarrow 1, * : 1$ and rewrite rule $f x \Rightarrow *$. This relation is confluent, but when taken together with η -contractions confluence is lost as equation 2 demonstrates — for a detailed discussion the reader should consult [3].

These deficiencies in η -contractions have recently led several authors [1, 2, 6, 12] to reconsider the old proposal [11, 13, 14] that η -conversion be interpreted as an expansion

$$f \Rightarrow \lambda x^T . f x \quad \text{if } f : T \rightarrow T' \text{ and } x \notin \mathbf{FV}(f)$$

and the resulting rewrite relation has been shown confluent. In these works infinite reduction sequences such as

$$f \Rightarrow \lambda x^T . f x \Rightarrow \lambda x^T . (\lambda y^T . f y) x \Rightarrow \dots$$

are avoided by imposing syntactic restrictions to limit the possibilities for expansion; namely λ -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* [11, 14]. Most pleasingly of all, these properties tend to be maintained if one adds other type constructors [7], or algebraic rewrite rules [3].

In addition to these practical arguments, the category-theoretic analysis of reduction [7] provides another argument in favour of interpreting η as an expansion. In this analysis the introduction and elimination rules of a type constructor form a pair of locally adjoint functors whose local unit and counit are respectively an expansionary (not *contractive*) η -rewrite rule and contractive β -rewrite rule. The associated local triangle laws assert the existence of looping reductions — for the exponential the triangle laws are

$$\begin{array}{lcl} \lambda x.t & \Rightarrow & \lambda y.(\lambda x.t)y \quad \Rightarrow \quad \lambda y.t[y/x] \equiv \lambda x.t \\ tu & \Rightarrow & (\lambda x.tx)u \quad \Rightarrow \quad tu \end{array} \quad (3)$$

Thus even the restrictions on η -expansion required to obtain strong normalisation have a categorical formulation, preventing exactly those expansions occurring in the triangle laws 3.

This categorical approach to rewriting has been extended to the more difficult problem of providing a decision procedure for $\beta\eta$ -equality for type constructors such as the coproduct, the tensor of linear logic and even the !-type constructor of linear logic. [8, 7]. Again, the introduction and elimination rules for these type constructors are deemed to form an adjoint pair with rewrite rules being derived from the associated unit and counit. However, the resulting η -rewrite rules for these type constructors are substantially more complex than those for the product and exponential — 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 are eliminated. This leads to some difficult term rewriting problems — the interested reader may consult the above references.

This paper extends the initial results in a different direction by investigating the use of expansionary η -rewrite rules in a polymorphic λ -calculus called System F [10, 9]. This calculus was introduced by Girard over twenty years ago and may be thought of as the simply typed λ -calculus enriched with type variables and a mechanism for forming Π -types by universally quantifying over all other types. Elements of these Π -types are thought of as polymorphic functions and there are introduction and elimination rules which describe how polymorphic functions may be defined and how such polymorphic functions may be used to construct other functions.

After presenting System F, we define an equational theory called *$\beta\eta$ -equality* on the terms of System F by adding second order β - and η -equations to their usual first order counterparts. Next, the restrictions on the applicability of the first order η -expansions are generalised to the second order η -expansions and we obtain

a rewrite relation which we prove to be strongly normalising, confluent, and to have as its reflexive, symmetric and transitive closure $\beta\eta$ -equality. We conclude by showing that the normal forms of this rewrite relation are exactly the second order long $\beta\eta$ -normal forms.

Our proof of strong normalisation of the restricted rewrite relation is a cross between the traditional proof of strong normalisation for the fragment containing only β -reductions [10], and the proof of strong normalisation for the simply typed λ -calculus with expansionary η -rewrite rules [12, 7]. This requires several alterations to the traditional definition a *reducibility candidate* to cope with the presence of expansions. This reducibility candidate method differs from the modular approach of [5] which investigates rewriting in System F but does not consider the second order η -rewrite rule. The reader is encouraged to consult [4] where a similar approach to rewriting in System F to that of this paper is taken.

In conclusion, there are many reasons for using η -expansions in type theory, e.g. the ease with which they generalise to other type constructors and their good modularity properties. This paper shows that η -expansions are also robust enough to be applied to polymorphic type theories where until recently the only possibility was to use contractive η -rewrite rules. This paper is thus an important first step towards the smooth combination of expressive type theories with $\beta\eta$ -equality and algebraic rewrite systems.

2 System F

Although the reader will be assumed to be familiar with System F, we shall give a brief presentation of the calculus for completeness. The formulation of System F presented here is based on that found in [10] and takes advantage of the relatively simple type structure of System F to avoid a presentation based on contexts. Of course there are still a few technical details about which we must be careful, but the overall simplification of notation is considerable.

Let \mathbf{Var}^* be an infinite set of *type variables*. The *types* of System F are defined as by the grammar

$$T := X \mid T \rightarrow T \mid \Pi X.T$$

where $X \in \mathbf{Var}^*$. The set of all types of System F is denoted $\Lambda(*)$ and a type is called *atomic* iff it is a member of \mathbf{Var}^* . We use $T, U, V, ..$ to range over types and $X, Y, Z, ...$ to range over atomic types. The set of free type variables occurring in a type T is denoted $\mathbf{FTV}(T)$ and α -equivalent types are treated as being equal. A type valued substitution is a finite partial function $\theta : \mathbf{Var}^* \rightarrow \Lambda(*)$ and the result of applying such a substitution to a type T is defined as expected and denoted $T\theta$.

There is also an infinite set of *term variables*, \mathbf{Var} , which is disjoint from \mathbf{Var}^* .

These term variables are used to construct the *pre-terms* of System F as follows:

$$t := x^T \mid tt \mid \lambda x^T.t \mid tT \mid \Lambda X.t$$

where x is a term variable, T is a type and X is a type variable. The set of pre-terms of System F is denoted Λ and we use t, u, v, \dots to range over pre-terms and x, y, z, \dots to range over term variables. The following definitions are used throughout the paper:

- A pre-term is an *introduction* pre-term iff it is of the form $\lambda x^T.t$ or $\Lambda X.t$.
- A pre-term is *neutral* if it is not an introduction term.
- A sub-preterm is said to occur *negatively* in a pre-term iff it is either applied to another pre-term or applied to a type.
- In the pre-term x^T , the type T is called the *type annotation* of the variable x . Sometimes the type annotation will be omitted so as to increase legibility.

The free term variables of a pre-term t are denoted $\mathbf{FV}(t)$, while the free type variables of a pre-term t are denoted $\mathbf{FTV}(t)$. We now present inference rules for assigning types to pre-terms and, in order to avoid the notationally cumbersome use of contexts, we assume a function $\phi : \mathbf{Var} \rightarrow \Lambda(*)$ which assigns to each variable a unique type. For each type T we assume $\mathbf{Var}(T) = \phi^{-1}(T)$ is infinite and variables are sometimes written with a type annotation to indicate their type. The *typing judgements* of System F are of the form $t:T$, where t is a pre-term and T is a type, and are derived by the inference rules in Table 1. Terms differing only

Table 1: Typing Judgements for System F

| | |
|-------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------|
| $\frac{x \in \mathbf{Var}(T)}{x^T : T}$ | |
| $\frac{t : T' \quad x \in \mathbf{Var}(T)}{\lambda x^T.t : T \rightarrow T'}$ | $\frac{t : T \rightarrow T' \quad u : T}{tu : T'}$ |
| $\frac{t : \Pi X.T \quad U \text{ is a type}}{tU : T[U/X]}$ | $\frac{t : T \quad x^U \in \mathbf{FV}(t) \text{ implies } X \notin \mathbf{FTV}(U)}{\Lambda X.t : \Pi X.T}$ |

in bound term variables of the same type and bound type variables are identified. A pre-term t is said to be a *term* iff there is a type T and a typing judgement $t:T$, and in this case we shall say that t has type T . The set of terms which have type T is denoted $\Lambda(T)$. A term-valued substitution is a type-indexed partial function $\theta_T : \mathbf{Var}(T) \rightarrow \Lambda(T)$ such that $\theta_T(x) \neq x^T$ for only a finite set of types

T and variables $x \in \mathbf{Var}(T)$. The result of applying a term-valued substitution θ to a term t is denoted $t\theta$ and defined as expected. A type valued substitution θ is only applied to a term t iff $x^T \in \mathbf{FV}(t)$ implies $\mathbf{dom}(\theta) \cap \mathbf{FTV}(T) = \emptyset$ and may implicitly require renaming of bound variables.

Lemma 2.1 *The following are true*

- *If there is a typing judgement $t:T$ and $x^T \in \mathbf{FV}(t)$, then $\mathbf{FTV}(T) \subseteq \mathbf{FTV}(t)$*
- *If there are typing judgements $t:T$ and $t:T'$, then $T = T'$.*
- *If θ is a type substitution, and there is a judgement $t : T$, then there is also a judgement $t\theta : T\theta$*
- *If θ is a term valued substitution and there is a term judgement $t : T$, then there is also a typing judgement $t\theta : T$.*

Proof Induction on the typing judgement $t : T$ □

Later on we shall permit the η -expansion of a term providing the term inhabits a function type or a universally quantified type and hence the decidability of type inhabitation will be crucial in enumerating the reducts of a term. Fortunately in our formulation of System F typeability and type inhabitation are decidable properties.

Lemma 2.2 *Given a pre-term t it is decidable whether there exists a type T such that there is a typing judgement $t : T$. In addition, given a pre-term t and a type T it is decidable whether there is a typing judgement $t : T$.*

Proof Both parts of the lemma are proved by induction on the structure of t . □

3 $\beta\eta$ -equality System F

Equality is usually defined in System F as the least congruence on terms containing the following pair of basic equations

$$\begin{array}{ll} \beta_{\rightarrow} & (\lambda x^T.t)u = t[u/x] \\ \beta_{\Pi} & (\Lambda X.t)V = t[V/X] \end{array} \quad (4)$$

The rewrite relation obtained by orienting these equations from left to right is denoted \Rightarrow_{β} and is well-known to be both strongly normalising and confluent [10]. However this is a rather minimal equational theory and in the rest of this paper we consider the effect of adding first and second order η -equations.

The equational theory known as $\beta\eta$ -equality is the least congruence on the terms of System F including the equations of Table 2. Although $\beta\eta$ -equality has been defined on the set of all terms, the next lemma shows that $\beta\eta$ -equality is actually a family of equational theories indexed by the types of System F.

Table 2: $\beta\eta$ -Equality in System F

| | | | |
|---------------------|--------------------|-----------------|-----------------------------------------------------------------------------------------------|
| $\beta \rightarrow$ | $(\lambda x^T.t)u$ | $=_{\beta\eta}$ | $t[u/x]$ |
| $\eta \rightarrow$ | t | $=_{\beta\eta}$ | $\lambda x^T.tx^T \quad x \notin \mathbf{FV}(t) \text{ and } t \in \Lambda(T \rightarrow T')$ |
| β_{Π} | $(\Lambda X.t)V$ | $=_{\beta\eta}$ | $t[V/X]$ |
| η_{Π} | t | $=_{\beta\eta}$ | $(\Lambda X.tX) \quad X \notin \mathbf{FTV}(t) \text{ and } t \in \Lambda(\Pi X.T)$ |

Lemma 3.1 *Assume $t : T$ and $u : U$ are terms such that $t =_{\beta\eta} u$. Then $T = U$.*

Proof One need only consider the equations of Table 2, and the lemma is proved by considering the typing derivation of the left hand side of each equation. \square

Bearing in mind the discussion in the introduction concerning the problems associated with the use of η -contractions, we are naturally led to investigate $\beta\eta$ -equality in System F by using β -contractions and η -expansions of the form

$$\frac{t : T \rightarrow T' \quad x \notin \mathbf{FV}(t)}{t \Rightarrow \lambda x^T.tx^T} \quad \frac{t : \Pi X.T \quad X \notin \mathbf{FV}(t)}{t \Rightarrow \Lambda X.tX}$$

As with the first order η -rewrite rule, unrestricted use of second order η -expansions permits infinite reduction sequences. In addition to the first order reduction loops of equation 3, there are the following second order reduction loops:

$$\begin{array}{lcl} \Lambda X.t & \Rightarrow & \Lambda Y.(\Lambda X.t)Y \Rightarrow \Lambda Y.t[Y/X] \equiv \Lambda X.t \\ tU & \Rightarrow & (\Lambda X.tX)U \Rightarrow tU \end{array} \quad (5)$$

We follow [8, 7] in defining a rewrite relation $\Rightarrow_{\mathcal{F}}$ by placing restrictions on when these expansions are permitted. In fact it turns out that the obvious generalisation of the first order restrictions suffice to obtain a strongly normalising and confluent rewrite relation whose equational theory is $\beta\eta$ -equality. Thus λ -abstractions and Λ -abstractions may not be η -expanded and nor may terms which are applied to other terms or to types. These latter, context sensitive, restrictions on expansion are enforced by simultaneously defining a further subrelation $\Rightarrow_{\mathcal{T}}$ of $\Rightarrow_{\mathcal{F}}$ which is guaranteed not to include top-level expansions, i.e. rewrites of the form $t \Rightarrow \lambda x^T.tx^T$ and $t \Rightarrow \Lambda X.tX$. Thus, a negatively occurring subterm may be safely $\Rightarrow_{\mathcal{T}}$ -rewritten without the risk of creating reduction loops as in equations 3 and 5. First, define a function mapping terms to terms

$$\eta(t) = \begin{cases} t & \text{if } t \text{ is of atomic type} \\ \lambda x^T.tx^T & \text{if } t : T \rightarrow T', \quad x \notin \mathbf{FV}(t) \\ \Lambda X.tX & \text{if } t : \Pi X.T, \quad X \notin \mathbf{FTV}(t) \end{cases}$$

A term is *expandable* iff it is neutral and of non-atomic type. The inference rules in Table 3 simultaneously define a relation $\Rightarrow_{\mathcal{F}}$, called the *restricted rewrite relation*,

and another relation $\Rightarrow_{\mathcal{I}}$ which is the ‘internal’ counterpart of $\Rightarrow_{\mathcal{F}}$. By lemma 2.2, the expandability of a term is decidable and hence the $\Rightarrow_{\mathcal{I}}$ and $\Rightarrow_{\mathcal{F}}$ -reducts of a term are enumerable.

Table 3: The Restricted Rewrite Relation

$$\begin{array}{c}
\frac{t \Rightarrow_{\beta} t'}{t \Rightarrow_{\mathcal{I}} t'} \quad \frac{t \text{ is expandable}}{t \Rightarrow_{\mathcal{F}} \eta(t)} \quad \frac{t \Rightarrow_{\mathcal{I}} t'}{t \Rightarrow_{\mathcal{F}} t'} \\
\\
\frac{t \Rightarrow_{\mathcal{I}} t'}{tV \Rightarrow_{\mathcal{I}} t'V} \quad \frac{t \Rightarrow_{\mathcal{F}} t'}{\Lambda X.t \Rightarrow_{\mathcal{I}} \Lambda X.t'} \\
\\
\frac{t \Rightarrow_{\mathcal{I}} t'}{tu \Rightarrow_{\mathcal{I}} t'u} \quad \frac{u \Rightarrow_{\mathcal{F}} u'}{tu \Rightarrow_{\mathcal{I}} tu'} \quad \frac{t \Rightarrow_{\mathcal{F}} t'}{\lambda x^T.t \Rightarrow_{\mathcal{I}} \lambda x^T.t'}
\end{array}$$

Lemma 3.2 *The least equivalence relation containing $\Rightarrow_{\mathcal{F}}$ is precisely $\beta\eta$ -equality.*

Proof If $t \Rightarrow_{\mathcal{F}} t'$, then clearly $t =_{\beta\eta} t'$. In addition, if t' is obtained from t by a prohibited expansion, then as equations 3 and 5 show, there is always a β -reduction, and hence an $\Rightarrow_{\mathcal{F}}$ -rewrite $t' \Rightarrow_{\mathcal{F}} t$. Thus the smallest equivalence relation containing $\Rightarrow_{\mathcal{F}}$ is $\beta\eta$ -equality. \square

When proving strong normalisation we shall require the following lemmas. Note that since $\Rightarrow_{\mathcal{F}}$ is not a congruence, these lemmas are not as trivial as they may first appear

Lemma 3.3 *Let tx^T and uX be terms. If $x^T \notin \text{FV}(t)$ and tx^T is $\Rightarrow_{\mathcal{F}}$ -strongly normalising, then t is also $\Rightarrow_{\mathcal{F}}$ -strongly normalising. Similarly, if $X \notin \text{FTV}(u)$ and uX is $\Rightarrow_{\mathcal{F}}$ -strongly normalising, then u is also $\Rightarrow_{\mathcal{F}}$ -strongly normalising.*

Proof We prove by induction on the normalisation rank of tx^T that all the one-step reducts of t are $\Rightarrow_{\mathcal{F}}$ -strongly normalising. This implies that t is $\Rightarrow_{\mathcal{F}}$ -strongly normalising. The one-step reducts are

- $t \Rightarrow_{\mathcal{F}} \lambda x^T.tx^T$. The term $\lambda x^T.tx^T$ is $\Rightarrow_{\mathcal{F}}$ -strongly normalising because all reduction sequences of this term are induced by reduction sequences of the term tx^T which by assumption is $\Rightarrow_{\mathcal{F}}$ -strongly normalising.
- $t \Rightarrow_{\mathcal{I}} t'$. In this case there is a reduction $tx^T \Rightarrow_{\mathcal{I}} t'x^T$ and so $t'x^T$ is $\Rightarrow_{\mathcal{F}}$ -strongly normalising. By induction, this means that t' is $\Rightarrow_{\mathcal{F}}$ -strongly normalising.

A similar argument holds for the second half of the lemma. \square

The restrictions imposed on the applicability of η -expansions prevent the expansion of introduction terms and subterms which occur negatively. However it is possible to “fake” such expansions as the following reduction sequence shows:

$$(\lambda x^{T \rightarrow U} .x)(\lambda y^T .y) \Rightarrow_{\mathcal{I}} (\lambda x^{T \rightarrow U} .\eta(x))(\lambda y^T .y) \Rightarrow_{\mathcal{I}} \eta(\lambda y^T .y)$$

and

$$(\lambda x^{T \rightarrow U} .xy)z \Rightarrow_{\mathcal{I}} (\lambda x^{T \rightarrow U} .xy)\eta(z) \Rightarrow_{\mathcal{I}} \eta(z)y$$

Even though we seem to have “smuggled in” an expansion of a λ -abstraction or a negatively occurring subterm, these fake expansions have required a β -reduction and so cannot be used to construct infinite reduction sequences. Formalising this idea amounts to an analysis of the interaction between substitution and η -expansion. A relation is *substitutive* if whenever there are reductions $t \Rightarrow t'$ and $u \Rightarrow u'$ then there is also a reduction sequence $t[u/x] \Rightarrow^* t'[u'/x]$. Of course the rewrite relation $\Rightarrow_{\mathcal{F}}$ is not a congruence and so $\Rightarrow_{\mathcal{F}}$ -reduction is not substitutive. The next lemma characterises when substitutivity fails and, in these instances, exhibits alternative reduction sequences which suffice for our needs:

Lemma 3.4 *Let V be a type and t, t', u, 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 an introduction term and t' is obtained by expanding an occurrence of x in t . In this case there are reduction sequences $t[\eta(u)/x] \Rightarrow_{\beta}^* t'[u'/x] \Rightarrow_{\beta}^* t[u/x]$.*
- *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 case $t[u'/x]$ and $t[u/x]$ have a common $\Rightarrow_{\mathcal{I}}^*$ -reduct.*
- *There is a rewrite $t[V/X] \Rightarrow_{\mathcal{R}} t'[V/X]$.*

Proof The three parts of the lemma are proved separately by induction on the rewrite in question. The first part follows because if u is an introduction term, then $\eta(u) \Rightarrow_{\beta}^* u$, while the reduct mentioned in the second part of the lemma is constructed from $t[u/x]$ by expanding those instances of x in t which do not occur negatively and then substituting u . The final part of the lemma holds as type substitutions preserve the neutrality of a term and also the non-atomicity of the type of the term. \square

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, e.g. there are the following

counterexamples:

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

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 3.5 *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 for t_2 .*

Proof The proof is by simultaneous induction on the term t , with the tricky cases handled by lemma 3.4. \square

4 A Proof of Strong Normalisation for $\Rightarrow_{\mathcal{F}}$

Our proof of strong normalisation of the relation $\Rightarrow_{\mathcal{F}}$ is a cross between the traditional proof of strong normalisation for β -reduction, e.g. see [10], and the proof of strong normalisation for the simply typed λ -calculus with expansionary η -rewrite rules [7, 12]. Thus, for every type we shall define a predicate, called a *reducibility candidate*, on sets of terms of that type. The set of $\Rightarrow_{\mathcal{F}}$ -strongly normalising terms of an atomic type form a reducibility candidate of that type and are used to construct canonical reducibility candidates of higher types. We prove that these canonical reducibility candidates contain all terms of that type and, as a corollary, conclude that all terms are $\Rightarrow_{\mathcal{F}}$ -strongly normalising.

As with the first order case, the definition of reducibility candidate must be altered from that used to prove strong normalisation of \Rightarrow_{β} -reduction so as to cope with the presence of expansionary rewrites. These alterations come in two parts: (i) the predicate (CR3) is weakened so that the η -expansion of a neutral term need not be considered and; (ii) a new predicate is introduced to ensure that reducibility candidates are closed under η -expansion. It has been suggested that these predicates are too strong, in particular that there is no need for CR4. However, all alternative proofs I have seen require something like lemma 4.2 to be proven and these proofs forget that reduction is not substitutive and hence fail to properly re-establish the induction hypothesis. The only solution is to make use of lemma 3.4 and this requires that the η -expansion of a reducible term is reducible — hence the need for CR4.

Formally, a *reducibility candidate* of type U is a set P of terms of type U which satisfy the following four reducibility predicates:

CR1 If $t \in P$ then t is $\Rightarrow_{\mathcal{F}}$ -strongly normalising.

CR2 If $t \in P$ and $t \Rightarrow_{\mathcal{I}} t'$ then $t' \in P$.

CR3 If t is a neutral term and all $\Rightarrow_{\mathcal{I}}$ -reducts of t are members of P , then $t \in P$.

CR4 If $t \in P$ then $\eta(t) \in P$.

The set of reducibility candidates of type U is denoted $\mathbf{RC}(U)$ and \mathbf{RC} denotes the set of all reducibility candidates. Let $|_| : \mathbf{RC} \rightarrow \Lambda(*)$ be the function which maps a reducibility candidate to the unique type of the terms which belong to it. If $t \in \mathcal{S}$ for some reducibility candidate \mathcal{S} , then the term t is called \mathcal{S} -reducible — when \mathcal{S} is clear from the context we simply say t is reducible. Define

$$\mathbf{SN}(T) = \{t \in \Lambda(T) \mid t \text{ is } \Rightarrow_{\mathcal{F}}\text{-strongly normalising}\}$$

Lemma 4.1 *If X is an atomic type, then $\mathbf{SN}(X) \in \mathbf{RC}(X)$, while if $\mathcal{S} \in \mathbf{RC}(T)$ and $x \in \mathbf{Var}(T)$ is a term variable, then $x^T \in \mathcal{S}$.*

Proof We must establish that the set of terms $\mathbf{SN}(X)$ satisfies the four reducibility predicates. CR1 is a tautology, while if t is $\Rightarrow_{\mathcal{F}}$ -strongly normalising and $t \Rightarrow_{\mathcal{I}} t'$ then, as $\Rightarrow_{\mathcal{I}}$ is a subrelation of $\Rightarrow_{\mathcal{F}}$, t' is also $\Rightarrow_{\mathcal{F}}$ -strongly normalising. CR3 holds because all $\Rightarrow_{\mathcal{I}}$ -reducts of a term t are $\Rightarrow_{\mathcal{F}}$ -strongly normalising by assumption and, because t is a term of atomic type, t has no η -expansion and so no other reducts. Finally, CR4 also holds because terms of atomic type have no η -expansion.

The second part of the lemma follows from the reducibility predicate CR3 because variables are neutral terms and have no $\Rightarrow_{\mathcal{I}}$ -reducts. Hence any reducibility candidate must contain all variables of that type. \square

By lemma 4.1 we know that for atomic types X , $\mathbf{RC}(X) \neq \emptyset$. These reducibility candidates are now used to construct reducibility candidates of higher type.

4.1 Exponentials

Let $\mathcal{R} \in \mathbf{RC}(U)$ and $\mathcal{S} \in \mathbf{RC}(V)$ be reducibility candidates. Define the set of terms

$$\mathcal{R} \rightarrow \mathcal{S} = \{t \in \Lambda(U \rightarrow V) \mid \forall u \in \mathcal{R}. tu \in \mathcal{S}\}$$

Before proving that $\mathcal{R} \rightarrow \mathcal{S}$ is a reducibility candidate, we give an alternate characterisation of which λ -abstractions are members of $\mathcal{R} \rightarrow \mathcal{S}$:

Lemma 4.2 *Let $\mathcal{R} \in \text{RC}(U)$ and $\mathcal{S} \in \text{RC}(V)$ be reducibility candidates. If $x \in \text{Var}(U)$ and for all $u \in \mathcal{R}$, the term $t[u/x] \in \mathcal{S}$, then $\lambda x^U.t \in (\mathcal{R} \rightarrow \mathcal{S})$.*

Proof By lemma 4.1 $x^U \in \mathcal{R}$ and hence by assumption $t[x^U/x] \in \mathcal{S}$. Thus by CR2 for \mathcal{S} , t is $\Rightarrow_{\mathcal{F}}$ -strongly normalising. We prove by induction on the sum of the $\Rightarrow_{\mathcal{F}}$ -normalisation ranks of t and u that if $u \in \mathcal{R}$, then $(\lambda x^U.t)u$ is reducible. By CR3 one need only show that all $\Rightarrow_{\mathcal{I}}$ -reducts of $(\lambda x^U.t)u$ are reducible. The one-step $\Rightarrow_{\mathcal{I}}$ -reducts of $(\lambda x^U.t)u$ induced by rewrites of u are reducible by induction, while given a rewrite $t \Rightarrow_{\mathcal{F}} t'$ we must re-establish the induction hypothesis. Given any \mathcal{R} -reducible term v , by lemma 3.4 there is a reduction of at least one of the following forms

$$t[v/x] \Rightarrow_{\mathcal{F}}^* t'[v/x] \text{ or } t[\eta(v)/x] \Rightarrow_{\mathcal{F}}^* t'[v/x]$$

Thus $t'[v/x]$ is reducible and so $\lambda x^U.t'$ satisfies the induction hypothesis. Thus, by induction, $(\lambda x^U.t')u$ is reducible and, as the only other $\Rightarrow_{\mathcal{I}}$ -reduct is $t[u/x]$, which is reducible by assumption, the lemma is proved. \square

We can now prove that $\mathcal{R} \rightarrow \mathcal{S}$ is a reducibility candidate.

Lemma 4.3 *If $\mathcal{R} \in \text{RC}(U)$ and $\mathcal{S} \in \text{RC}(V)$, then the set of terms $\mathcal{R} \rightarrow \mathcal{S}$ is a reducibility candidate.*

Proof We shall establish the four properties.

CR1 By lemma 4.1, if x is a term variable of type U , then x is reducible. Thus, if $t \in \mathcal{R} \rightarrow \mathcal{S}$, then tx^U is reducible and so tx^U is $\Rightarrow_{\mathcal{F}}$ -strongly normalising. Thus by lemma 3.3, t is also $\Rightarrow_{\mathcal{F}}$ -strongly normalising.

CR2 Let $t \in \mathcal{R} \rightarrow \mathcal{S}$ and $t \Rightarrow_{\mathcal{I}} t'$. Then for any $u \in \mathcal{R}$, $tu \in \mathcal{S}$ and $tu \Rightarrow_{\mathcal{I}} t'u$. Thus $t'u \in \mathcal{S}$ and hence $t' \in \mathcal{R} \rightarrow \mathcal{S}$.

CR3 Induction on the $\Rightarrow_{\mathcal{F}}$ -normalisation rank of u is used to prove that if $u \in \mathcal{R}$ then $tu \in \mathcal{S}$. Because t is neutral, the one-step $\Rightarrow_{\mathcal{I}}$ -reducts of tu are either of the form $t'u$ where $t \Rightarrow_{\mathcal{I}} t'$ or tu' where $u \Rightarrow_{\mathcal{F}} u'$. The first class of terms are \mathcal{S} -reducible because by assumption $t' \in \mathcal{R} \rightarrow \mathcal{S}$, while terms of the latter form are \mathcal{S} -reducible by induction.

CR4 Let $t \in (\mathcal{R} \rightarrow \mathcal{S})$. By lemma 4.2 we must prove that if $u \in \mathcal{R}$ then $(tx^U)[u/x] = tu$ is \mathcal{S} -reducible. But this is exactly the definition of $\mathcal{R} \rightarrow \mathcal{S}$. \square

4.2 Universally Quantified types

Let T be a type. A *reducibility parameter* for T consists of a partial function $\theta : \text{Var}^* \rightarrow \text{RC}$ such that the free type variables of T are contained in the domain of θ . Every reducibility has an underlying type-valued substitution $|\theta| : \text{Var}^* \rightarrow \Lambda(*)$

which maps a variable $X \in \mathbf{dom}(\theta)$ to the type underlying the reducibility candidate $\theta(X)$. Given a reducibility parameter θ for T , define the set of terms $T\theta$ as

- If $T = X$ then $T\theta = \theta(X)$
- If $T = U \rightarrow V$ then $T\theta = U\theta \rightarrow V\theta$
- If $T = \Pi Y.W$ then

$$T\theta = \bigcap_{V \in \Lambda(*)} \bigcap_{\mathcal{S} \in \mathbf{RC}(V)} \{t \in T \mid \theta \mid tV \in W\theta[Y \mapsto \mathcal{S}]\}$$

where $\theta[Y \mapsto \mathcal{S}]$ is the function θ , except that Y is mapped to \mathcal{S} .

Before showing that $T\theta$ is a reducibility candidate, we prove the analogue of lemma 4.2 for universal types.

Lemma 4.4 *Let θ be a reducibility parameter for $\Pi Y.W$ such that for every reducibility candidate $\mathcal{S} \in \mathbf{RC}(V)$, $W\theta[Y \mapsto \mathcal{S}]$ is a reducibility candidate and $w[V/Y] \in W\theta[Y \mapsto \mathcal{S}]$. Then $\Lambda Y.w \in (\Pi Y.W)\theta$.*

Proof We have to show that $(\Lambda Y.w)V \in W\theta[Y \mapsto \mathcal{S}]$ for every type V and reducibility candidate \mathcal{S} . By lemma 4.1, $\mathbf{SN}(Y) \in \mathbf{RC}(Y)$ and so $w \in W\theta[Y \mapsto \mathbf{SN}(Y)]$. Thus w is $\Rightarrow_{\mathcal{F}}$ -strongly normalising and the lemma may be proved using the reducibility predicate CR3 and by induction on the $\Rightarrow_{\mathcal{F}}$ -normalisation rank of w . The one-step $\Rightarrow_{\mathcal{I}}$ -reducts of $(\Lambda Y.w)V$ are $w[V/Y]$ and terms of the form $(\Lambda Y.w')V$ where $w \Rightarrow_{\mathcal{F}} w'$. The first term is reducible by assumption while by lemma 3.4 w' satisfies the induction hypothesis and so $\Lambda Y.w'$ is a member of $(\Pi Y.W)\theta$. Hence $\Lambda Y.w \in (\Pi Y.W)\theta$. \square

Lemma 4.5 *If θ is a reducibility parameter for T , then $T\theta \in \mathbf{RC}(T \mid \theta)$.*

Proof The proof is by induction on the type T . If T is a type variable the lemma is trivial, while if T is an exponential the lemma follows by induction and lemma 4.3. The only case left is where $T = \Pi Y.W$.

CR1 Let $t \in T\theta$. As $\mathbf{SN}(Y) \in \mathbf{RC}(Y)$, $tY \in W\theta[Y \mapsto \mathbf{SN}(Y)]$. Hence tY is $\Rightarrow_{\mathcal{F}}$ -strongly normalising and thus by lemma 3.3 so is t .

CR2 If $t \in T\theta$ and $t \Rightarrow_{\mathcal{I}} t'$, then for all reducibility candidates $\mathcal{S} \in \mathbf{RC}(V)$, $tV \Rightarrow_{\mathcal{I}} t'V$ and so, by CR2, $t'V \in W\theta[Y \mapsto \mathcal{S}]$. Hence $t' \in T\theta$.

CR3 Let t be a neutral term, all of whose $\Rightarrow_{\mathcal{I}}$ -reducts are $T\theta$ -reducible. The one step $\Rightarrow_{\mathcal{I}}$ -reducts of tV are of the form $t'V$ where $t \Rightarrow_{\mathcal{I}} t'$ and so, for any reducibility candidate $\mathcal{S} \in \mathbf{RC}(V)$, $t'V \in W\theta[Y \mapsto \mathcal{S}]$. Thus by CR3 we conclude that tV is also $W\theta[Y \mapsto \mathcal{S}]$ -reducible and hence $t \in T\theta$.

CR4 Let t be a member of $(\Pi Y.W)\theta$. By lemma 4.4, if we can prove that for all reducibility candidates $\mathcal{S} \in \text{RC}(V)$, $(tY)[V/Y] = tV$ is $W\theta[Y \mapsto \mathcal{S}]$ -reducible, then $(\Lambda Y.tY) \in (\Pi Y.W)\theta$. But this follows as $t \in (\Pi Y.W)\theta$.

□

In proving strong normalisation we shall need the following lemma relating these constructions of reducibility candidates from reducibility parameters to type-valued substitutions.

Lemma 4.6 *Let θ be a reducibility parameter for T and V . Then the reducibility candidates*

$$(T[V/Y])\theta \quad \text{and} \quad T\theta[Y \mapsto V\theta]$$

are equal.

Proof The proof is by induction on the type T .

□

Before proving that all terms are strongly normalising we find alternate criteria for proving a term is a member of a given reducibility candidate.

Lemma 4.7 *If $t \in (\Pi Y.W)\theta$, then if θ is also a reducibility parameter for V , $tV \in (W[V/Y])\theta$.*

Proof By hypothesis $tV \in W\theta[Y \mapsto \mathcal{S}]$ for every reducibility candidate $\mathcal{S} \in \text{RC}(V)$. The lemma follows by taking \mathcal{S} to be $V\theta$ and using lemma 4.6.

□

Strong normalisation is a corollary to proving that the substitution of reducible terms into a term produces a reducible term. As there are two types of variable quantified over in System F this substitution must be defined on two levels.

Theorem 4.8 *Let t be a term of type T . Suppose the free term variables of t are amongst \vec{x} which have types \vec{U} and that θ is a reducibility parameter for T . If u_1, \dots, u_n are $U_1\theta, \dots, U_n\theta$ -reducible, then $t|\theta|[\vec{u}/\vec{x}] \in T\theta$. Thus all terms are $\Rightarrow_{\mathcal{F}}$ -strongly normalising.*

Proof The proof of the first part of the lemma is by induction over the structure of t and follows the standard procedure. The second part is proved by instantiating the first part with $\theta(X) = \text{SN}(X)$ and $u_i = x_i$ so as to obtain t is reducible and hence by CR1 $\Rightarrow_{\mathcal{F}}$ -strongly normalising.

□

Theorem 4.9 *The relation $\Rightarrow_{\mathcal{F}}$ is confluent and hence each term has a unique $\Rightarrow_{\mathcal{F}}$ -normal form.*

Proof We have established local confluence in lemma 3.5 and strong normalisation in theorem 4.8. These properties imply confluence. In addition, confluence and strong normalisation imply that every term has a unique normal form.

□

5 Further Results

We collect here some further results pertaining to the use of η -expansions in System F. First, we fulfill our earlier promise of showing that a term is a $\Rightarrow_{\mathcal{F}}$ -normal form iff it is a long $\beta\eta$ -normal form. A term is a *long $\beta\eta$ -normal form* [11] iff it is a β -normal form and all subterms are either of base type, introduction terms, occur negatively or are actually types. A term is an *internal long $\beta\eta$ -normal form* iff it is a β -normal form and all subterms, apart from the term itself, are either of base type, introduction terms, occur negatively or are types

Lemma 5.1 *A term is a long $\beta\eta$ -normal form iff it is a $\Rightarrow_{\mathcal{F}}$ -normal form.*

Proof Both directions of the lemma are easily established by induction on the structure of t , while simultaneously proving that a term is a $\Rightarrow_{\mathcal{I}}$ -normal form iff it is an internal long $\beta\eta$ -normal form. \square

One can calculate the $\Rightarrow_{\mathcal{F}}$ -normal form of a term by contracting all β -redexes and then performing any remaining η -expansions

Lemma 5.2 *The $\Rightarrow_{\mathcal{F}}$ -normal form of a term may be calculated by first contracting all β -redexes and then performing any remaining expansions.*

Proof The proof rests on showing that if t is a β -normal form and $t \Rightarrow_{\mathcal{F}} t'$ then t' is also β -normal form. But this is easily done as all η -expansions which create new β -redexes are prohibited by the restrictions on expansions. \square

In fact one can go further and give an explicit function which defines the η -normal form of a term. The first step is to characterise the reducts of a variable and this is done by the function Δ

$$\begin{aligned} \Delta(z^X) &= \{z\} \\ \Delta(z^{T \rightarrow U}) &= \{z\} \cup \{\lambda x^T.v[zu/y] \mid u \in \Delta(x^T) \text{ and } v \in \Delta(y^U)\} \\ \Delta(z^{\Pi X.T}) &= \{z\} \cup \{\lambda X.v[zX/y] \mid v \in \Delta(y^T)\} \end{aligned}$$

Lemma 5.3 *If z is a term variable, then $z^T \Rightarrow_{\mathcal{F}}^* \alpha$ iff $\alpha \in \Delta(z^T)$*

Proof The two containments are established by induction over type structure. Certainly there is a reduction sequence

$$z^{T \rightarrow U} \Rightarrow_{\mathcal{F}} \lambda x^T.zx \Rightarrow_{\mathcal{I}}^* \lambda x^T.zu \Rightarrow_{\mathcal{I}}^* \lambda x^T.v[zu/y]$$

for any $u \in \Delta(x^T)$ and $v \in \Delta(y^U)$. Note that the second reduction sequence follows from the induction hypothesis, while the third reduction sequence also makes use of lemma 3.4 since zu is not an introduction term. A similar argument holds for variables of Π -types.

The reverse containment is established by showing that $\Delta(z)$ is closed under

reduction. For variables of base type this is trivially true, while any reduct of $\lambda x^T.v[zu/y]$ is easily shown to be induced by reductions of u or v . Again, a similar argument holds for Π -types. \square

Now let $\Delta^m(z^T)$ be the largest member of $\Delta(z)$. Define functions η^I and η^F which map terms to terms as follows

Table 4: Definition of The Eta-long Form Of A Term

| | | | |
|---------------------------|---|----------------------------|--------------------------------------------|
| $\eta^I(x^T)$ | = | x^T | |
| $\eta^I(tu)$ | = | $\eta^I(t)\eta^F(u)$ | |
| $\eta^I(tU)$ | = | $\eta^I(t)U$ | |
| $\eta^I(\lambda x : A.t)$ | = | $\lambda x : A.\eta^F(t)$ | |
| $\eta^I(\Lambda X.t)$ | = | $\Lambda X.\eta^F(t)$ | |
| $\eta^F(t)$ | = | $\eta^I(t)$ | if $t : T$ and t is an introduction term |
| | = | $\Delta(z^T)[\eta^I(t)/z]$ | otherwise |

Lemma 5.4 *There are reduction sequences $t \Rightarrow_{\mathcal{I}}^* \eta^I(t)$ and $t \Rightarrow_{\mathcal{F}}^* \eta^F(t)$. In addition, if t is a β -normal form, then $\eta^I(t)$ is an $\Rightarrow_{\mathcal{I}}$ -normal form while $\eta^F(t)$ is an $\Rightarrow_{\mathcal{F}}$ -normal form.*

Proof The first half of the lemma is a simple induction on type structure, while the second follows by induction on term structure and lemma 5.2. \square

6 Future Research

In this paper we have shown that η -expansions may successfully be applied to Girard's System F. There are two principle directions in which this research may be extended.

Firstly one may further increase the expressiveness of the type theory by adding higher order polymorphism such as found in F^ω , and type dependency as found in the Calculus of Constructions. Applying η -expansions to these type theories requires the solution of new technical problems. In the case of F^ω , terms no longer inhabit unique types and so one needs to be careful in formalising the idea of expanding a term if it inhabits a function/polymorphic type. Things are even more complex in the Calculus of Constructions as types have terms embedded within them. As a result, one can no longer assume the decidability of type equality, typeability and type inhabitation.

The second major area of research is to justify the claims that algebraic rewrite systems can be smoothly added to type theories containing η -expansion. In

particular, the standard techniques in the literature should suffice to show that if we add a confluent or strongly normalising algebraic rewrite system to System F with η -expansions then the combined system retains these key properties.

Of course, the ultimate goal is to eventually combine these lines of research by showing how η -expansions can be used to smoothly combine algebraic rewrite systems in powerful type theories such as the Calculus of Constructions.

7 Acknowledgments

I am grateful to Dr R. Di Cosmo with whom I have had many informative conversations and who has also spotted flaws in earlier works. I would also like to thank the Royal Society of London who funded a visit to Paris where much of this research was carried out. Finally Don Sannella has helped enormously.

References

- [1] Y. Akama. On Mints' reduction for ccc-calculus. In *Typed λ -Calculus and Applications*, volume 664 of *Lecture Notes in Computer Science*, pages 1–12. Springer Verlag, 1993.
- [2] R. Di Cosmo and D. Kesner. Simulating expansions without expansions. *Mathematical Structures in Computer Science*, 4:1–48, 1994.
- [3] R. Di Cosmo and D. Kesner. Combining algebraic rewriting, extensional λ -calculi and fixpoints. In *TCS*, 1995.
- [4] R. Di Cosmo and D. Kesner. Rewriting with polymorphic extensional λ -calculus. 1995.
- [5] R. Di Cosmo and A. Piperno. Expanding extensional polymorphism. In *Typed λ -calculus and Applications*, volume 902 of *Lecture Notes in Computer Science*, pages 139–153. Springer Verlag, 1995.
- [6] D. Dougherty. Some λ -calculi with categorical sums and products. In *Rewriting Techniques and Applications*, volume 690 of *Lecture Notes in Computer Science*, pages 137–151. Springer Verlag, 1993.
- [7] N. Ghani. *Adjoint Rewriting*. PhD thesis, University of Edinburgh, Department of Computer Science, 1995.
- [8] N. Ghani. $\beta\eta$ -equality for coproducts. In *Typed λ -calculus and Applications*, number 902 in *Lecture Notes in Computer Science*, pages 171–185. Springer Verlag, 1995.

- [9] J. Y. Girard. The system F of variable types - 15 years later. *Theoretical Computer Science*, 45:159–192, 1986.
- [10] J. Y. Girard, P. Taylor, and Y. Lafont. *Proofs and Types*, volume 7 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 1989.
- [11] G. Huet. *Résolution d'équations dans des langages d'ordre 1, 2, ..., ω* . Thèse d'Etat, Université de Paris VII, 1976.
- [12] C. B. Jay and N. Ghani. The virtues of eta-expansion. *Journal of Functional Programming*, Volume 5(2), April 1995, pages 135-154. CUP 1995.
- [13] G. E. Mints. Teorija kategorii i teorija dokazatelstv.i. *Aktualnye problemy logiki i metodologii nauky*, pages 252–278, 1979.
- [14] D. Prawitz. Ideas and results in proof theory. In J.E. Fenstad, editor, *Proc. 2nd Scandinavian Logic Symposium*, pages 235–307. North Holland, 1971.