Categorical reconstruction of a *reduction free* normalization proof ⋆.

Thorsten Altenkirch[1], Martin Hofmann[2] ⋆⋆ and Thomas Streicher[3]

[1] Chalmers University of Technology, Göteborg, Sweden
email : alti@cs.chalmers.se
[2] University of Edinburgh, Scotland
email : mxh@dcs.ed.ac.uk
[3] Technische Hochschule Darmstadt, Germany
email : streicher@mathematik.th-darmstadt.de

1 Introduction

We present a categorical proof of the normalization theorem for simply typed λ-calculus, i.e. we derive a computable function nf which assigns to every typed λ-term a normal form, s.t.

$$\frac{M \simeq N}{\mathrm{nf}(M) = \mathrm{nf}(N)} \qquad \mathrm{nf}(M) \simeq M$$

where $\simeq$ is $\beta\eta$ equality. Both the function nf and its correctness properties can be deduced from the categorical construction. To substantiate this, we present an ML program in the appendix which can be extracted from our argument.

We emphasize that this presentation of normalization is *reduction free*, i.e. we do not mention term rewriting or use properties of term rewriting systems such as the Church-Rosser property. An immediate consequence of normalization is the decidability of $\simeq$ but there are other useful corollaries; for instance we can show that every closed term M with the type $(o \to o) \to (o \to o)$ is $\beta\eta$-equivalent to a Church numeral of the form $\lambda xy.x^i y$ for some number i.

1.1 Related work

Already Martin-Löf in [ML75] shows the basic normalization property of Type Theory by a model construction and without referring to properties of term rewriting. However, he considers only weak equality, i.e. excluding the ξ-rule. Schwichtenberg and Berger in [BS91] give a reduction free normalization proof which is based on the idea to *invert the evaluation functional.* Our presentation can be seen as an attempt towards a rational reconstruction of their proof using

⋆ A preliminary version of this work has been presented at the joint CLICS/TYPES workshop on *Categories and Type Theory* in Göteborg, January 1995

⋆⋆ Supported by a EU HCM fellowship, contract number ERBCHBICT930420.

the language of category theory[4] . Another outcome of categorical normalisation proofs like ours is a certain clarification of the role of variables in usual normalisation arguments based on Tait's computability method. Recall that in these proofs a key step consists of applying a functional term to *a variable* the origin of which is rather opaque because variables do not correspond to elements in the set-theoretic interpretation. In the presheaf interpretation variables show up naturally as generalized elements.

T. Coquand and Dybjer present an analysis of reduction free normalization for the weak case in [CD93]. They observe that the correctness proof is related to categorical glueing, without, however, making the relationship precise. C. Coquand in [Coq94] presents a complete formalization of a reduction-free normalization proof in Martin-Löf's intuitionistic set theory using the ALF system. It is worthwhile to note that she makes the use of Kripke style models explicit.

A naive category-theoretic approach to intensional aspects like normalisation faces two problems. First, in category-theory $\beta\eta$-equal terms are usually identified and so we cannot express the difference between any term and a term in normal form. Second, if category theory is understood via classical set theory then a category-theoretic construction need not give rise to an algorithm. We overcome the first obstacle by modeling normal forms (and the related concept of neutral terms) by a special presheaf (over the category of variable renamings) of normal forms based. The second problem is solved by restricting ourselves to those parts of category theory which can be formalised in a constructive setting, for definiteness in extensional Martin-Löf type theory with subset types as described in [NPS90]. It seems worthwhile to carry out the formalization of category theory in a type theoretic setting, this is a topic of current research, e.g. [HS95].

1.2 An intuitionistic completeness proof

C. Coquand, T. Coquand and Dybjer have observed that there is a close analogy between intuitionistic completeness proofs and normalization. Indeed, there is an intriguing relationship between an intuitionistic completeness proof for Kripke-style semantics of minimal logic which we are going to sketch here and our construction. The following proof is folklore and inspired by similar constructions which can be found in the standard literature, e.g. [TvD88].

Minimal intutionistic logic can be presented by the following proof system where A is a purely implicational formula and Γ is a finite set of such formulas :

$$\frac{A \in \Gamma}{\Gamma \vdash A} \qquad \frac{\Gamma \cup \{A\} \vdash B}{\Gamma \vdash A \to B} \qquad \frac{\Gamma \vdash A \to B \qquad \Gamma \vdash A}{\Gamma \vdash B}$$

This can be extended to an entailment relation between contexts $\Gamma \vdash \Delta$.

[4] Note, however, that it is not clear that the algorithm we derive is identical to theirs. E.g. Berger and Schwichtenberg are using the `eval` function in LISP, which does not exist in ML (nor in Category Theory).

A (propositional) Kripke structure (c.f. [Kri65]) $(W, \leq, \Vdash)$ is a preordered set of worlds $(W, \leq)$ together with a *forcing relation* $\Vdash$ between worlds and propositional constants such that for all propositional constants A the following monotonicity criterion is satisfied : $w' \Vdash A$ if $w \leq w'$ and $w \Vdash A$.

The forcing relation $\Vdash$ can be extended to compound formulas and contexts by the following inductive definition :

$$w \Vdash A \to B :\Longleftrightarrow \forall_{w' \geq w} w' \Vdash A \Rightarrow w' \Vdash B$$
$$w \Vdash \Gamma :\Longleftrightarrow w \Vdash A \text{ for all } A \in \Gamma$$

It is easy to show that the monotonicity condition extends to compound formulas and contexts.

We say that Γ entails A ($\Gamma \models A$) in the structure if any world w forces A whenever w forces Γ :

$$\Gamma \models A :\Longleftrightarrow \forall_w w \Vdash \Gamma \Rightarrow w \Vdash A$$

By a straightforward induction over the structure of derivation one can show soundness, i.e. $\Gamma \vdash A$ implies $\Gamma \models A$ for all structures.

Completeness means that the reverse implication holds as well, i.e. if $\Gamma \models A$ holds in all Kripke structures then also $\Gamma \vdash A$. To show this we construct a special structure, the universal model. Here the worlds are the contexts and $\leq$ is the subset relation. For atomic propositions we set

$$\Gamma \Vdash A :\Longleftrightarrow \Gamma \vdash A$$

It is easy to see that monotonicity holds. The essential step in the completeness proof is to establish by induction over the structure of formulas that $\Gamma \Vdash A$ iff $\Gamma \vdash A$. Note that we have to show both directions simultaneously. Completeness is obtained by noting that $\Gamma \vdash \Gamma$ and using the equivalence above.

What happens to this construction when we apply the Curry-Howard isomorphism? We suggest the following translation :

Minimal logic $\sim$ CCCs
Kripke structure $\sim$ Presheaf category
Preorder of worlds $\sim$ base category
Soundness $\sim$ Presheaves are cartesian closed

The main point of this consideration is that the completeness proof can be extended to this setting and gives rise to a normalisation function.

2 Contextual CCCs

Simply typed λ-calculus is usually identified with cartesian closed categories (CCCs). This can be formalised as a categorical equivalence between the two notions [LS86]. A consequence of this equivalence is that the initial cartesian closed category can be construed from the syntax of typed lambda calculus by taking contexts as objects and substitutions as morphisms. However, since in an arbitrary CCC products are not associative up to equality, this category is only initial up to isomorphism, but not in the strict sense. From a category theorist's point of view this does not matter, but for syntactic considerations, as we pursue them, it is important to have actual equality. Therefore we define a generalization of cartesian closed categories which is closer to the syntax and for which the term model is initial in the strict sense.

Definition 1. **A contextual CCC C** is given by

1. a category $\mathbf{C}$
2. A (chosen) terminal object $\mathbf{1}$,
3. A full subcategory $\mathbf{T}$ of types,
4. For each $\Gamma \in |\mathbf{C}|$ and $A \in |\mathbf{T}|$ a (chosen) cartesian product $\Gamma \times A$ in $\mathbf{C}$.
5. For every $A, B \in |\mathbf{T}|$ a (chosen) exponential of B by A, that is an object $A \Rightarrow B \in |\mathbf{T}|$ together with a natural isomorphism:

$$\mathbf{C}(- \times A, B) \simeq \mathbf{C}(-, A \Rightarrow B)$$

Fact 1 *Every CCC gives rise to a contextual CCC with the choice* $\mathbf{T} = \mathbf{C}$ *and product and exponentials inherited from the cartesian closed structure.*

Similar to [LS86] we shall use an equational presentation of contextual CCCs . We introduce the following constants : [5]

$$\begin{aligned}
!_\Gamma &\in \mathbf{C}(\Gamma, \mathbf{1}) \\
< -, - > &\in \mathbf{C}(\Delta, \Gamma) \times \mathbf{C}(\Delta, A) \to \mathbf{C}(\Delta, \Gamma \times A) \\
0_{\Gamma,A} &\in \mathbf{C}(\Gamma \times A, A) \\
\pi_{\Gamma,A} &\in \mathbf{C}(\Gamma \times A, \Gamma) \\
\Lambda(-) &\in \mathbf{C}(\Gamma \times A, B) \to \mathbf{C}(\Gamma, A \Rightarrow B) \\
\mathrm{APP}(-, -) &\in \mathbf{C}(\Gamma, A \Rightarrow B) \times \mathbf{C}(\Gamma, A) \to \mathbf{C}(\Gamma, B)
\end{aligned}$$

which fulfill the following equations :

$$\begin{aligned}
!_A &= \gamma \qquad \gamma \in \mathbf{C}(\Gamma, \mathbf{1}) \\
0 \circ < \gamma, f > &= f \\
\pi \circ < \gamma, f > &= \gamma \\
< \pi \circ \gamma, 0 \circ \gamma > &= \gamma \\
\mathrm{APP}(\Lambda(f), a) &= f \circ < 1, a > \\
\Lambda(\mathrm{APP}(f \circ \pi, 0)) &= f
\end{aligned}$$

[5] As said in the Introduction all the category-theoretic and set-theoretic constructions we make are understood in a constructive setting. Without fixing a particular such setting, typically extensional Martin-Löf Type Theory, we use standard set-theoretic notation like $\times$ (cartesian product), $\to$ (function space) $x \in A \mapsto \ldots$ (abstraction), $\{x \in A \mid P\}$ (separation), etc., to denote expressions in such a constructive set theory.

2.1 The free contextual CCC

Let $\mathbf{B}$ be a small category (of base types and constants). A contextual CCC over $\mathbf{B}$ is a contextual category $(\mathbf{C}, \mathbf{T})$ together with a functor $F : \mathbf{B} \to \mathbf{T}$. We can construct the *free contextual CCC* over $\mathbf{B}$, this is the initial object in the category of contextual CCCs over $\mathbf{B}$ which has as morphisms functors that preserve the CCC-structure on the nose and commute with the Fs.

In the following we shall restrict our attention to the case where $\mathbf{B}$ is the category $\mathbf{1}$ with one object and one arrow (i.e. the terminal object in $\mathbf{CAT}$). The free contextual CCC generated by $\mathbf{1}$ will be denoted by $\mathbf{Tm}$. It corresponds to the simply typed λ-calculus with one uninterpreted base type, which we denote by o. By initiality, a contextual CCC $\mathbf{C}$ with a chosen object $X \in |\mathbf{C}|$ gives rise to a unique strict structure preserving functor :

$$[\![-]\!]^X \in \mathbf{Tm} \to \mathbf{C}$$

such that $[\![o]\!]^X = X$.

The free contextual CCC over $\mathbf{1}$ $\mathbf{Tm}$ can be constructed following the equational presentation above, which is equivalent to a de Bruijn-style presentation of simply typed λ-calculus (therefore we use 0 to denote the second projection). To emphasize the syntactic character of the morphisms in $\mathbf{Tm}$ we use $(), (-,-), \lambda(-), \mathrm{app}(-,-)$ for $!, < -, - >, \Lambda, \mathrm{APP}$ in $\mathbf{Tm}$. One question which arises naturally is whether the equality of syntactically constructed morphisms in $\mathbf{Tm}$ is decidable.

2.2 The category of weakenings

Our model construction is based on the category of presheaves over a category of context extensions which we are going to define. These morphisms generalise the notion of subsequence on contexts used in the proof-irrelevant case in 1.2. We could restrict attention to composites of the π-morphisms and thereby arrive at a posetal category. For technical reasons it is, however, appropriate to include morphism like $<!, 0_{\mathbf{1}\times A, A}>: (\mathbf{1} \times A) \times A \to \mathbf{1} \times A$.

We present the category W of weakenings by an inductive definition, together with a faithful embedding E into Tm.

Definition 2. We define a family of sets $\mathrm{W}(-,-)$ indexed by objects in $\mathbf{Tm}$ inductively by the following constructors :

$$\begin{aligned} 1_\Gamma &\in \mathrm{W}(\Gamma, \Gamma) \\ \mathrm{w}_1(-) &\in \mathrm{W}(\Gamma, \Delta) \to \mathrm{W}(\Gamma \times A, \Delta) \\ \mathrm{w}_2(-) &\in \mathrm{W}(\Gamma, \Delta) \to \mathrm{W}(\Gamma \times A, \Delta \times A) \end{aligned}$$

and define composition $\circ$ and an embedding $\mathrm{E} \in \mathrm{W}(\Gamma, \Delta) \to \mathbf{Tm}(\Gamma, \Delta)$ by primitive recursion :

$$1 \circ w = w$$

$$\begin{aligned}
\mathrm{w}_1(w) \circ w' &= \mathrm{w}_1(w \circ w') \\
\mathrm{w}_2(w) \circ 1 &= \mathrm{w}_2(w) \\
\mathrm{w}_2(w) \circ \mathrm{w}_1(w') &= \mathrm{w}_1(w \circ w') \\
\mathrm{w}_2(w) \circ \mathrm{w}_2(w') &= \mathrm{w}_2(w \circ w')
\end{aligned}$$

$$\begin{aligned}
\mathrm{E}(1) &= 1 \\
\mathrm{E}(\mathrm{w}_1(w)) &= \mathrm{E}(w) \circ \pi \\
\mathrm{E}(\mathrm{w}_2(w)) &= (\mathrm{E}(w) \circ \pi, 0) \\
&= \mathrm{E}(w) \times 1
\end{aligned}$$

Proposition 3. W *with* 1 *and* $\circ$ *as defined above is a category, and* E *is a faithful embedding into* **Tm**.

Proof. By a simple structural induction.

Note that $\mathrm{E}(\mathrm{w}_1(1)) = \pi$ and therefore we also use π to denote this morphism in W.

3 Normalization

We shall present our construction in several steps : First we will sketch the interpretation of λ-calculus in presheaf categories which corresponds to a proof-relevant version of the soundness theorem for Kripke structures. Similarly a proof relevant version of the completeness proof can be constructed and indeed this gives rise to an inverse of the evaluation functor. However, not much is gained by this, since the resulting normalization function only works on equivalence classes and is hence extensionally equal to the identity. We introduce *normal form objects* to overcome this problem, but it is no longer obvious that proof object corresponding to completeness is inverse to the evaluation functor. Finally, we construct a new category **TwGl** (for twisted glueing) which combines a normalization function with its correctness proof. This category is a variation of the glueing construction and could be viewed as a special case of a *Kripke logical predicate* as described in [MM91]. [6]

3.1 Interpretation in presheaf categories

Given a small category **C** we construct the category of presheaves over **C**: $\mathrm{PSh}(\mathbf{C}) = \mathbf{Set}^{\mathbf{C}^{\mathrm{op}}}$, i.e. objects are functors into **Set** and morphisms are natural transformations. For objects $F \in |\mathrm{PSh}(\mathbf{C})|$ and $A \in |\mathbf{C}|$ we denote application by F_A and analogously for $\alpha \in \mathrm{PSh}(\mathbf{C})(F, G)$ we denote the instance at A by $\alpha_A \in F_A \to G_A$. Moreover given $x \in F_A$ and $w \in \mathbf{C}(B, A)$ we denote $F(w)(x) \in F_B$ by x^{+w}.

[6] Berger [Ber94] also found a new version of the correctness argument where he makes use of a logical predicate.

It is well known that PSh(C) is a CCC [7] and therefore also a contextual CCC :

Proposition 4. PSh(C) *is a CCC, with the following choice*

1. $\mathbf{1}_D^{\mathrm{PSh}(\mathbf{C})} = \mathbf{1}^{\mathbf{Set}}$
2. $(F \times^{\mathrm{PSh}(\mathbf{C})} G)_D = F_D \times G_D$
3.

$$(F \Rightarrow^{\mathrm{PSh}(\mathbf{C})} G)_D = \mathrm{Nat}(\mathbf{C}(-, D) \times F, G)$$

$$\Lambda^{\mathrm{PSh}(\mathbf{C})}(\alpha \in \mathrm{PSh}(\mathbf{C})(F \times G, H))_A \in F_A \to (G \Rightarrow^{\mathrm{PSh}(\mathbf{C})} H)_A$$

$$\Lambda^{\mathrm{PSh}(\mathbf{C})}(\alpha)_A(x)_B(w, y) = \alpha_B(x^{+w}, y)$$

$$\mathrm{APP}^{\mathrm{PSh}(\mathbf{C})}(\alpha \in \mathrm{PSh}(\mathbf{C})(F, G \Rightarrow^{\mathrm{PSh}(\mathbf{C})} H), \beta \in \mathrm{PSh}(\mathbf{C})(F, G))_A \in F_A \to H_A$$

$$\mathrm{APP}^{\mathrm{PSh}(\mathbf{C})}(\alpha, \beta)_A(x) = \alpha_A(x)(1_A, \beta_A(x))$$

We can view a presheaf category as a *proof relevant* Kripke structure: the base category **C** corresponds to the *opposite* of the preordered set of worlds. The forcing relation is generalized to a presheaf where the monotonicity corresponds to functoriality. Semantic entailment in a Kripke structure is generalized to a morphism in the presheaf category, i.e. by a natural transformation. The soundness theorem generalizes to the fact that for any $F \in \mathrm{PSh}(\mathbf{C})$ we obtain an evaluation functor $[\![-]\!]^F \in \mathbf{Tm} \to \mathrm{PSh}(\mathbf{C})$.

3.2 A proof relevant completeness proof

We can generalize the intuitionistic completeness proof for Kripke structures to the presheaf semantics, which gives rise to an inverse of the evaluation functor. This is our first step towards the normalization function.

The category corresponding to the universal model is PSh(W). To construct the evaluation functor we use

$$\mathrm{Yw} = \mathbf{Set}^{\mathrm{E}} \circ \mathrm{Y} \in \mathbf{Tm} \to \mathrm{PSh}(\mathrm{W})$$

where $\mathrm{Y}(A) = \mathbf{Tm}(-, A)$ is the Yoneda embedding. We obtain

$$[\![-]\!]^{\mathrm{Yw}(o)} \in \mathbf{Tm} \to \mathrm{PSh}(\mathrm{W})$$

[7] We never use the fact that PSh(C) is a topos, and indeed this cannot be shown, if we use a predicative set theory like Martin-Löf's. However, one *could* show that PSh(C) is a model of Martin-Löf's set theory (i.e. an LCCC with some additional structure) and express our construction directly in the internal language of this category.

Corresponding to the lemmas used in the completeness proof we construct the following families of maps for any $\Gamma, \Delta \in |\mathbf{Tm}|$:

$$\begin{aligned}
\mathrm{q}^1_{A,\Gamma} &\in [\![A]\!]^{\mathrm{Yw}(o)}_{\Gamma} \to \mathrm{Yw}(A)_{\Gamma} \\
\mathrm{q}^1_{o,\Gamma}(M) &= M \\
\mathrm{q}^1_{A\Rightarrow B,\Gamma}(f) &= \lambda(\mathrm{q}^1_{B,\Gamma\times A}(f_{\Gamma.A}(\pi_{\Gamma,A}, \mathrm{u}^1_{A,\Gamma\times A}(0_{\Gamma,A}))))
\end{aligned}$$

$$\begin{aligned}
\mathrm{u}^1_{A,\Gamma} &\in \mathrm{Yw}(A)\Gamma \to [\![A]\!]^{\mathrm{Yw}(o)}_{\Gamma} \\
\mathrm{u}^1_{o,\Gamma}(M) &= M \\
\mathrm{u}^1_{A\Rightarrow B,\Gamma}(M)_{\Delta}(w, x) &= \mathrm{u}^1_{B,\Gamma}(\mathrm{app}(M^{+w}, \mathrm{q}^1_{A,\Gamma}(x))
\end{aligned}$$

Note that $\mathrm{Yw}(\Gamma)\Delta = \mathbf{Tm}(\Delta, \Gamma)$.

The extensions to contexts $\mathrm{q}^1_{\Delta,\Gamma}, \mathrm{u}^1_{\Delta,\Gamma}$ is defined in an obvious componentwise way.

Here q^1 stands for quote since it maps semantical objects to syntax and u^1 for unquote because it is doing the converse.

We can define the proof relevant counterpart of the completeness proof thus obtaining :

$$\mathrm{compl}^1_{\Gamma,\Delta} \in \mathrm{PSh}(\mathbf{C})([\![\Gamma]\!], [\![\Delta]\!]) \to \mathrm{Tm}(\Gamma, \Delta)$$

$$\mathrm{compl}^1_{\Gamma,\Delta}(\gamma) = (\mathrm{q}^1_{\Delta,\Gamma} \circ \gamma_\Gamma \circ \mathrm{u}^1_{\Gamma,\Gamma})(1_\Gamma)$$

Consider the following diagram :

$$\begin{tikzcd}
\mathbf{1} \arrow[r, "1_\Gamma"] & \mathrm{Yw}(\Gamma)_\Gamma \arrow[r, "\mathrm{u}_{\Gamma,\Gamma}"'] \arrow[dr, "1"'] & {[\![\Gamma]\!]_\Gamma} \arrow[r, "{[\![M]\!]_\Gamma}"'] \arrow[d, "\mathrm{q}^1_{\Gamma,\Gamma}"] & {[\![\Delta]\!]_\Gamma} \arrow[d, "\mathrm{q}^1_{\Delta,\Gamma}"] \\
& & \mathrm{Yw}(\Gamma)_\Gamma \arrow[r, "\mathrm{Yw}(M)_\Gamma"'] & \mathrm{Yw}(\Delta)_\Gamma
\end{tikzcd}$$

Using that q^1 is natural and that $\mathrm{q}^1_\Gamma \circ \mathrm{u}^1_\Gamma = 1$ (which can be shown by induction over the structure of Γ) we obtain that $\mathrm{compl}^1([\![M]\!]^{\mathrm{Yw}(o)}) = M$, i.e. compl^1 is the inverse of the evaluation functor.

The proof relevant version of the completeness proof shows that $[\![-]\!]$ is faithful. One would hope that the composition of evaluation and completeness

$$\begin{aligned}
\mathrm{nf}^1(M) &= \mathrm{compl}^1([\![M]\!]^{\mathrm{Yw}(o)}) \\
\mathrm{nf}^1 &\in \mathbf{Tm}(\Gamma, \Delta) \to \mathbf{Tm}(\Gamma, \Delta)
\end{aligned}$$

gives rise to a normalization function. However, this is not yet achieved, since we are only reasoning up to $\beta\eta$-equivalence and in this sense the constructed function is merely the identity.

3.3 Normal form objects

The shortcoming of the previous construction was that it only works up to equality. We can overcome this by identifying an *normal form object* with a trivial equality in PSh(W) and using this instead of $\mathrm{Yw}(A)$.

We define families $\mathrm{NF}(-,-)$ (normal forms) and $\mathrm{NE}(-,-)$ (neutral terms) indexed over objects from **Tm** inductively as follows :

$$0 \in \mathrm{NE}(\Gamma \times A, A) \qquad \frac{i \in \mathrm{NE}(\Gamma, A)}{i+1 \in \mathrm{NE}(\Gamma \times B, A)}$$

$$\frac{M \in \mathrm{NE}(\Gamma, A \Rightarrow B) \qquad N \in \mathrm{NF}(\Gamma, A)}{\mathrm{app}_{\Gamma,A,B}(M, N) \in \mathrm{NE}(\Gamma, B)}$$

$$\frac{M \in \mathrm{NE}(\Gamma, o)}{M \in \mathrm{NF}(\Gamma, o)}$$

$$\frac{M \in \mathrm{NF}(\Gamma \times A, B)}{\lambda_{\Gamma,A,B}(M) \in \mathrm{NF}(\Gamma, A \Rightarrow B)}$$

The rules for products for NF and NE are the same as in the definition of $\mathrm{Tm}(-,-)$, we used the same syntax as for Tm and it should be obvious that there are embeddings

$$\mathrm{NF}(\Gamma, \Delta) \hookrightarrow \mathrm{Tm}(\Gamma, \Delta)$$

$$\mathrm{NE}(\Gamma, \Delta) \hookrightarrow \mathrm{Tm}(\Gamma, \Delta)$$

Our definition of normal form corresponds to the standard definition of long $\beta\eta$-normal forms, i.e. of λ-terms which contain no β-redexes and are maximally η-expanded. However, our motivation is slightly different, we require a definition of normal forms s.t. q and nf are injective and onto. Note that the neutral terms are not just auxiliary for the definition of normal forms but that they are the minimal domain for which u has to be defined.

It is straightforward to define weakening for NF and NE and therefore we obtain presheaves for every $\Gamma \in |\mathbf{Tm}|$:

$$\mathrm{NF}(\Gamma) = \mathrm{NF}(-,\Gamma) \in |\mathrm{PSh}(\mathrm{W})|$$
$$\mathrm{NE}(\Gamma) = \mathrm{NE}(-,\Gamma) \in |\mathrm{PSh}(\mathrm{W})|$$

Moreover, there are obvious natural embeddings from NF and NE to **Tm**

$$\mathrm{i}_A \in \mathrm{PSh}(\mathrm{W})(\mathrm{NF}(A), \mathrm{Yw}(A))$$

$$\mathrm{i}'_A \in \mathrm{PSh}(\mathrm{W})(\mathrm{NE}(A), \mathrm{Yw}(A))$$

which we will often omit from calculations.

We can now modify the previous construction, i.e. we use

$$[\![-]\!]^{\mathrm{NF}(o)} \in \mathrm{Tm} \to \mathrm{PSh}(\mathrm{W})$$

as evaluation function and we modify q^1 and u^1 by replacing the basic morphisms from **Tm** by their syntactic counterparts in NF, NE thus obtaining :

$$\begin{aligned} \mathrm{q}^2_\Gamma &\in \mathrm{PSh(W)}(\llbracket\Gamma\rrbracket^{\mathrm{NF}(o)}, \mathrm{NF}(\Gamma)) \\ \mathrm{u}^2_\Gamma &\in \mathrm{PSh(W)}(\mathrm{NE}(\Gamma), \llbracket\Gamma\rrbracket^{\mathrm{NF}(o)}) \end{aligned}$$

We now obtain $\mathrm{nf}^2 \in \mathbf{Tm}(\Delta, \Gamma) \to \mathrm{NF}(\Delta, \Gamma)$. Thus we obtain a function which maps terms in an equivalence class to a normal form but it is no longer clear that the function we obtained is essentially the identity, i.e. whether

$$\mathrm{i}(\mathrm{nf}^2(\mathbf{M})) = \mathbf{M}$$

We are not even able to state that q^2 is a natural transformation, since $\mathrm{NF}(-)$ is not functorial. We could try to show that $\mathrm{q}^2_\Gamma \doteq \mathrm{i} \circ \mathrm{q}^1_\Gamma$ is natural but any attempts to show this directly fail — it seems that one needs a construction as the one described in the next two sections.

3.4 Twisted glueing

We define a category **TwGl** from which we can obtain a normalization function together with its correctness proof by proving that **TwGl** is a contextual CCC. The proof of the central theorem we defer to the next section.

Definition 5. We define the category **TwGl** by the following data

Objects : An object Γ is given as a tuple $(\tilde{\Gamma}, (\!|\Gamma|\!), \mathrm{u}_\Gamma, \mathrm{q}_\Gamma)$:

$$\begin{aligned} \tilde{\Gamma} &\in |\mathbf{Tm}| \\ (\!|\Gamma|\!) &\in |\mathrm{PSh(W)}| \\ \mathrm{u}_\Gamma &\in \mathrm{PSh(W)}(\mathrm{NE}(\tilde{\Gamma}), (\!|\Gamma|\!)) \\ \mathrm{q}_\Gamma &\in \mathrm{PSh(W)}((\!|\Gamma|\!), \mathrm{NF}(\tilde{\Gamma})) \end{aligned}$$

s.t. $\mathrm{i}_\Gamma \circ \mathrm{q}_\Gamma \circ \mathrm{u}_\Gamma = \mathrm{i}'_\Gamma$

Morphisms : A morphism $\alpha \in \mathbf{TwGl}(\Gamma, \Delta)$ is given by a pair $((\!|\alpha|\!), \boldsymbol{\alpha})$:

$$\begin{aligned} (\!|\alpha|\!) &\in \mathrm{PSh(W)}((\!|\Gamma|\!), (\!|\Delta|\!)) \\ \boldsymbol{\alpha} &\in \mathbf{Tm}(\tilde{\Gamma}, \tilde{\Delta}) \end{aligned}$$

s.t. the following diagram commutes in PSh(W):

$$\begin{tikzcd} (\!|\Gamma|\!) \arrow[r, "(\!|\alpha|\!)"'] \arrow[d, "\mathrm{q}_\Gamma"] & (\!|\Delta|\!) \arrow[d, "\mathrm{q}_\Delta"] \\ \mathrm{NF}(\tilde{\Gamma}) \arrow[d, "\mathrm{i}_\Gamma"] & \mathrm{NF}(\tilde{\Delta}) \arrow[d, "\mathrm{i}_\Delta"] \\ \mathrm{Yw}(\tilde{\Gamma}) \arrow[r, "\mathrm{Yw}(\boldsymbol{\alpha})"'] & \mathrm{Yw}(\tilde{\Gamma}) \end{tikzcd}$$

Identity,composition are defined componentwise.

We now come to the central theorem :

Theorem 6.

1. **TwGl** *is a contextual CCC.*
2. *The functor* $\mathrm{P} \in \mathbf{TwGl} \to \mathbf{Tm}$ *defined as*

$$\mathrm{P}(\tilde{\Gamma}, (\!|\Gamma|\!), \mathrm{u}_\Gamma, \mathrm{q}_\Gamma) = \tilde{\Gamma}$$
$$\mathrm{P}((\!|\alpha|\!), \boldsymbol{\alpha}) = \boldsymbol{\alpha}$$

preserves the contextual CCC structure on the nose.

To define the evaluation functor we need an object to interpret o . We note that

$$X = (o, \mathrm{NF}(o), 1, 1)$$

is an object on **TwGl** and hence we have

$$\llbracket - \rrbracket^X \in \mathbf{Tm} \to \mathbf{TwGl}$$

We define compl and nf as before :

$$\begin{aligned} \mathrm{compl}_{\Gamma,\Delta} &\in \mathrm{PSh}(\mathrm{W})(\Delta, \Gamma) \to \mathrm{NF}(\Delta, \Gamma) \\ \mathrm{compl}_{\Gamma,\Delta}(\mathbf{M}) &= \mathrm{q}_{\Delta,\Gamma}(\llbracket \mathbf{M} \rrbracket_\Gamma(\mathrm{u}_{\Gamma,\Gamma}(1_\Gamma))) \\ \mathrm{nf}_{\Gamma,\Delta} &\in \mathbf{Tm}(\Delta, \Gamma) \to \mathrm{NF}(\Delta, \Gamma) \\ \mathrm{nf}_{\Gamma,\Delta}(\mathbf{M}) &= \mathrm{compl}_{\Gamma,\Delta}(\llbracket \mathbf{M} \rrbracket) \end{aligned}$$

We can now conclude :

Proposition 7.

$$\mathrm{i} \circ \mathrm{nf}_\Gamma = 1_{\mathrm{Yw}\Gamma}$$

Proof. Consider the following diagram :

$$\begin{tikzcd}
\mathbf{1} \arrow[r, "1_\Gamma"] & \mathrm{NE}(\Gamma)_\Gamma \arrow[r, "\mathrm{u}_{\Gamma,\Gamma}"'] \arrow[ddr, "\mathrm{i}'"'] & (\!|\Gamma|\!)_\Gamma \arrow[r, "(\!|M|\!)_\Gamma"'] \arrow[d, "\mathrm{q}_{\Gamma,\Gamma}"] & (\!|\Delta|\!)_\Gamma \arrow[d, "\mathrm{q}_{\Delta,\Gamma}"] \\
& & \mathrm{NF}(\Gamma)_\Gamma \arrow[d, "\mathrm{i}"] & \mathrm{NF}(\Delta)_\Gamma \arrow[d, "\mathrm{i}"] \\
& & \mathrm{Yw}(\Gamma)_\Gamma \arrow[r, "\mathrm{Yw}(M)_\Gamma"'] & \mathrm{Yw}(\Delta)_\Gamma
\end{tikzcd}$$

The triangle and the square commute by the defining property for objects in **TwGl**. The result follows by noting that

$$(\mathrm{Yw}(M)_\Gamma \circ \mathrm{i}')(1_\Gamma) = M$$

We can now derive the corollaries mentioned in the introduction :

Corollary 8.

1. $\dfrac{M \simeq N}{\mathrm{nf}(M) = \mathrm{nf}(N)} \qquad \mathrm{nf}(M) \simeq M$
2. *$\simeq$ is decidable.*
3. *For all $M \in \mathbf{Tm}(\mathbf{1}, (o \to o) \to (o \to o))$ there is an i s.t. $M \simeq \lambda xy.x^i y$.*

Proof. 1. Note that $\simeq$ is the equality in $\mathbf{Tm}$. The first part follows from the fact that the interpretation in $\mathbf{TwGl}$ is sound and the second from proposition 7.
2. Follows from 1. and the fact that we work in a constructive set theory.
3. By a simple induction over the structure of normal forms we can show that the normal forms have this property and hence it follows for terms by the second part of 1.

3.5 Proof of theorem 6

Products The definition of $\mathbf{1}$ and $\times$ can be done in an obvious component-wise fashion.

The exponential The essential problem is the definition of the exponential. For the following assume $A, B \in |\mathbf{TwGl}|$.

The definitions of

$$\begin{aligned} \mathrm{q}_{A\Rightarrow B,\Delta} &\in [\![A \Rightarrow B]\!]_\Delta \to \mathrm{NF}(A \Rightarrow B)_\Delta \\ \mathrm{u}_{A\Rightarrow B,\Delta} &\in \mathrm{NE}(A \Rightarrow B)_\Delta \to [\![A \Rightarrow B]\!]_\Delta \end{aligned}$$

follows the definition sin the previous sections (3.2,3.3). We first define $\mathrm{q}_{A\Rightarrow B}$ and $\mathrm{u}_{A\Rightarrow B}$ explicitly using $(\![A]\!) \Rightarrow (\![B]\!)$. Later we will define $(\![A \Rightarrow B]\!)$ as subpresheaf of $(\![A]\!) \Rightarrow (\![B]\!)$. Naturally we will have to show that $\mathrm{u}_{A\Rightarrow B}$ produces results in the subpresheaf, whereas $\mathrm{q}_{A\Rightarrow B}$ causes no problems.

Lemma 9. *$\mathrm{q}_{A\Rightarrow B}$ and $\mathrm{u}_{A\Rightarrow B}$ are natural in Δ and therefore morphisms in* $\mathrm{PSh}(\mathrm{W})$ *:*

$$\begin{aligned} \mathrm{q}_{A\Rightarrow B} &\in \mathrm{PSh}(\mathrm{W})((\![A]\!) \Rightarrow (\![B]\!), \mathrm{NF}(A \Rightarrow B)) \\ \mathrm{u}_{A\Rightarrow B} &\in \mathrm{PSh}(\mathrm{W})(\mathrm{NE}(A \Rightarrow B), (\![A]\!) \Rightarrow (\![B]\!)) \end{aligned}$$

Lemma 10.

$$\mathrm{q}_{A\Rightarrow B} \circ \mathrm{u}_{A\Rightarrow B} = i_{A\Rightarrow B}$$

Proof. Assume Δ and $M \in \mathrm{NE}(\Delta, A \Rightarrow B)$ we have :

$$\begin{aligned} &\mathrm{q}_{A\Rightarrow B,\Delta}(\mathrm{u}_{A\Rightarrow B,\Delta}(M)) \\ &= \lambda(\mathrm{q}_{B,\Delta.A}(\mathrm{u}_{B,\Delta.A}(\mathrm{app}(M^{+\pi_{\Delta,A}}, \mathrm{q}_{A,\Delta.A}(\mathrm{u}_{A,\Delta.A}(0_{\Delta,A})))))) \\ &= \lambda(\mathrm{app}(M^{+\pi_{\Delta,A}}, 0_{\Delta,A})) \\ &= M \end{aligned}$$

$(\![A \Rightarrow B]\!)$ is defined as a subpresheaf of the full function space in PSh(W). We give here the explicit set-theoretic definition first :

Definition 11.

$$(\![A \Rightarrow B]\!)_\Gamma = \left\{ f \in ((\![A]\!) \Rightarrow (\![B]\!))_\Gamma \middle| \begin{array}{c} \forall_{\Delta \in |\mathbf{Tm}|, w \in \mathrm{W}(\Delta,\Gamma), x \in (\![A]\!)_\Delta} \\ \mathrm{i}(\mathrm{q}_{B,\Delta}(f(\Delta, w, x)) = \\ \mathrm{app}_{\Delta,A,B}(\mathrm{i}(\mathrm{q}_{A \Rightarrow B,\Gamma}(f)^{+w}), \mathrm{i}(\mathrm{q}_{A,D}(x))) \end{array} \right\}$$

$(\![A \Rightarrow B]\!)$ can be characterized as the equalizer of the following two arrows :

$$(\![A]\!) \Rightarrow (\![B]\!) \underset{(\mathrm{q}_A \Rightarrow 1_{\mathrm{Y_W}(B)}) \circ \Lambda(\mathrm{app}(\mathrm{i},\mathrm{i})) \circ \mathrm{q}_{A \Rightarrow B}}{\overset{1_{[A]} \Rightarrow \mathrm{q}_B}{\rightrightarrows}} (\![A]\!) \Rightarrow \mathrm{NF}(A)$$

We have to show that $\mathrm{u}_{A \Rightarrow B}$ will always produce results in $(\![A \Rightarrow B]\!)$.

Lemma 12.

$$\mathrm{u}_{A \Rightarrow B} \in \mathrm{PSh(W)}(\mathrm{NE}(A \Rightarrow B), (\![A \Rightarrow B]\!))$$

i.e. we have to show that for any $\Delta, M \in \mathrm{NE}(\Delta, A \Rightarrow B)$: $f = \mathrm{u}_{A \Rightarrow B,\Delta}(M)$ full fills the condition from definition 11.

Proof. Assume $\Delta \in |\mathbf{Tm}|, w \in \mathrm{W}(\Delta, \Gamma), x \in (\![A]\!)_\Delta$:

$$\begin{aligned} &\mathrm{q}_{B,\Delta}(\mathrm{u}_{A \Rightarrow B,\Delta}(M)(\Delta, w, x)) \\ &= \mathrm{u}_{B,\Delta}(q_{B,\Delta}(\mathrm{app}(M^{+w}, \mathrm{q}_{A,\Delta}(x)) \\ &= \mathrm{app}(M^{+w}, \mathrm{q}_{A,\Delta}(x)) \qquad \text{property of } B \\ &= \mathrm{app}(\mathrm{q}_{A \Rightarrow B,\Gamma}(\mathrm{u}_{A \Rightarrow B,\Gamma}(M))^{+w}, \mathrm{q}_{A,\Delta}(x)) \qquad \text{by lemma 10} \end{aligned}$$

We can conclude that

$$A \Rightarrow B = (\tilde{A} \Rightarrow \tilde{B}, (\![A \Rightarrow B]\!), \mathrm{u}_{A \Rightarrow B}, \mathrm{q}_{A \Rightarrow B})$$

is an object in **TwGl**. It remains to show that this is indeed an exponential.

Lemma 13. *Given $\alpha \in \mathbf{TwGl}(\Gamma \times A, B)$*

$$\begin{aligned} \Lambda^{\mathbf{TwGl}}(\alpha) &= (\Lambda^{\mathrm{PSh}(W)}((\![\alpha]\!)), \lambda(\boldsymbol{\alpha})) \\ &\in \mathbf{TwGl}(\Gamma, A \Rightarrow B) \end{aligned}$$

Proof. We have to verify :

1. The diagram from definition 5 commutes :

$$\mathrm{i} \circ \mathrm{q}_{A \Rightarrow B} \circ \Lambda^{\mathrm{PSh}(W)}(\llbracket \alpha \rrbracket) = \mathrm{Yw}(\lambda(\boldsymbol{\alpha})) \circ \mathrm{q}_{\Gamma}$$

Assume $\Delta, \gamma \in \llbracket \Gamma \rrbracket_{\Delta}$.

$$\begin{aligned}
&\mathrm{q}_{A\Rightarrow B}(\Lambda(\llbracket \alpha \rrbracket)_{\Delta}(\gamma)) \\
&= \lambda(\mathrm{q}_B(\llbracket \alpha \rrbracket_{\Delta \times A}(\gamma^{+\pi}, \mathrm{u}_{A,\Delta\times A}(0)))) \\
&= \lambda(\boldsymbol{\alpha} \circ \mathrm{q}_{\Gamma \times A, \Delta}(\gamma^{+\pi}, \mathrm{u}_{A,\Delta\times A}(0))) \qquad \text{property of } \alpha \\
&= \lambda(\boldsymbol{\alpha} \circ (\mathrm{q}_{\Gamma,\Delta}(\gamma^{+\pi}), \mathrm{q}_{A,\Delta}(\mathrm{u}_{A,\Delta\times A}(0)))) \\
&= \lambda(\boldsymbol{\alpha} \circ (\mathrm{q}_{\Gamma,\Delta}(\gamma^{+\pi}), 0) \qquad \text{by lemma 10} \\
&= \lambda(\boldsymbol{\alpha}) \circ \mathrm{q}_{\Gamma,\Delta}(\gamma)
\end{aligned}$$

2. $\Lambda^{\mathbf{TwGl}}(\alpha) \in \llbracket A \Rightarrow B \rrbracket$, that is we have to check the condition from definition 11 : Assume $\Delta, \gamma \in \llbracket \Gamma \rrbracket_{\Delta}, w \in \mathrm{W}(\Delta', \Delta), a \in \llbracket A \rrbracket_{\Delta'}$:

$$\begin{aligned}
&\mathrm{app}(\mathrm{q}_{A\Rightarrow B,\Delta'}(\Lambda(\llbracket \alpha \rrbracket)_{\Delta}(\gamma))^{+w}, \mathrm{q}_{A,\Delta'}(a)) \\
&= \mathrm{app}((\lambda(\boldsymbol{\alpha}) \circ \mathrm{q}_{\Gamma,\Delta}(\gamma))^{+w}, \mathrm{q}_{A,\Delta'}(a)) \qquad \text{previous case} \\
&= \mathrm{app}(\lambda(\boldsymbol{\alpha} \circ (\mathrm{q}_{\Gamma,\Delta}(\gamma^{+w})^{+\pi}, 0)), \mathrm{q}_{A,\Delta'}(a)) \\
&= \boldsymbol{\alpha} \circ (\mathrm{q}_{\Gamma,\Delta}(\gamma)^{+w}, \mathrm{q}_{A,\Delta'}(a)) \qquad (\beta) \\
&= \mathrm{q}_{\tilde{B},\Delta'}(\llbracket \alpha \rrbracket(\gamma^{+w}, a)) \qquad \text{property of } \alpha \\
&= \mathrm{q}_{\tilde{B},\Delta'}(\Lambda(\llbracket \alpha \rrbracket)'_{\Delta}(\gamma)(\Delta, w, a))
\end{aligned}$$

Lemma 14. *Assume* $\alpha \in \mathbf{TwGl}(\Gamma, A \Rightarrow B), \beta \in \mathbf{TwGl}(\Gamma, A)$

$$\begin{aligned}
\mathrm{APP}^{\mathbf{TwGl}}(\alpha, \beta) &= \mathrm{APP}^{\mathrm{PSh}(W)}((\llbracket \alpha \rrbracket, \llbracket \beta \rrbracket), \mathrm{app}(\boldsymbol{\alpha}, \boldsymbol{\beta})) \\
&\in \mathbf{TwGl}(\Gamma, B)
\end{aligned}$$

Proof. We only have to verify that he diagram from definition 5 commutes :

$$\mathrm{i} \circ \mathrm{q}_B \circ \mathrm{APP}^{\mathrm{PSh}(W)}(\llbracket \alpha \rrbracket, \llbracket \beta \rrbracket) = \mathrm{Yw}(\mathrm{app}(\boldsymbol{\alpha}, \boldsymbol{\beta})) \circ \mathrm{i} \circ \mathrm{q}_{\Gamma}$$

Assume $\Delta, \gamma \in \llbracket \Gamma \rrbracket_D$.

$$\begin{aligned}
&\mathrm{app}(\boldsymbol{\alpha}, \boldsymbol{\beta}) \circ (\mathrm{q}_{\Gamma,\Delta}(\gamma)) \\
&= \mathrm{app}(\boldsymbol{\alpha} \circ (\mathrm{q}_{\Gamma,\Delta}(\gamma)), \boldsymbol{\beta} \circ (\mathrm{q}_{\Gamma,\Delta}(\gamma))) \\
&= \mathrm{app}(\mathrm{q}_{A\Rightarrow B,\Delta}(\llbracket \alpha \rrbracket_{\Delta}(\gamma)), \mathrm{q}_A(\llbracket \beta \rrbracket_{\Delta}(\gamma))) \qquad \text{property of } \alpha \\
&= \mathrm{q}_{B,\Delta}(\llbracket \alpha \rrbracket_D(\gamma)(\Delta, 1_{\Delta}, \llbracket \beta \rrbracket_{\Delta}(\gamma))) \qquad \text{cond. from def.11} \\
&= \mathrm{q}_{B,\Delta}(\mathrm{APP}(\llbracket \alpha \rrbracket, \llbracket \beta \rrbracket)_{\Delta}(\gamma))
\end{aligned}$$

Lemma 15. $\llbracket A \Rightarrow B \rrbracket$ *is the exponential in* **TwGl** *with* $\Lambda^{\mathbf{TwGl}}$ *and* $\mathrm{APP}^{\mathbf{TwGl}}$.

Proof. By lemmas 13 and 14 we know that $\Lambda^{\mathbf{TwGl}}$ and $\mathrm{APP}^{\mathbf{TwGl}}$ are morphisms with the appropriate type. This is already sufficient since the equations follow from the corresponding equations on the components.

Proof of Theorem 6:

1. That **TwGl** is a contextual CCC follows from lemma 5 together with the trivial construction of products (which we have omitted).
2. That P preserves the contextual CCC structure on the nose follows immediately from the construction, since we defined products and exponential on the appropriate components directly by using the corresponding constructions from **Tm**.

□

4 Conclusions and further work

This work can be seen as a further attempt to demystify normalization proofs by freeing them from the syntactic details which bare the view to the underlying ideas. As a side effect we obtain new implementations of normalization, which may be interesting from a computational point of view.

We hope to be able to apply these ideas to obtain provably correct normalization functions for some systems for which the syntactical proofs become very hard or fail. The extension to products with surjective pairing seems rather trivial since we have already exploited the product structure in PSh(W) for the interpretation of contexts. A more challenging example is the normalization for strong coproducts which has only recently been investigated by Ghani [Gha95]. Following a suggestion by Thierry Coquand we are investigating the completeness proof for a variant of Beth models which gives rise to a normalization function for coproducts. The proof relevant version of those models are motivated by sheaf models; however it seems to be necessary to consider some modifications to the standard sheaf semantics.

Another interesting case are systems with dependent types where the fact that the η-rule leads to a failure of Church-Rosser on typed terms causes problems. We hope to be able to extend our approach to this case, thereby obtaining an explanation or an alternative to [Coq91].

Acknowledgements

This work has benefitted from numerous discussions with our colleagues in the BRAs CLICS (Categorical Logic in Computer Science) and TYPES (Types for Proofs and Programs). We would like to thank especially Ulrich Berger, Thierry and Catarina Coquand, Peter Dybjer, Andrew Pitts and Helmut Schwichtenberg. To typeset the diagrams we used Paul Taylor's diagram package.

A The normalization function in ML

The following ML program implements the function which can be extracted from our proof, i.e. we derive a function

```
val nf = fn : Ty list -> Ty -> Tm -> Tm
```

which, when applied to a context `G`, a type `S` and a term `M` with the property that `M` has type `S` in context `G`, calculates the long $\beta\eta$-normal form of `M`. If the algorithm is applied to incorrect arguments it may fail or may not terminate.

Note the following deviations from the proof :

- We do not introduce NF, NE but calculate a term. Although it would be possible to express the restriction to NF and NE in ML, this does not seem to be sensible from a programming point of view.
- We only normalize terms whereas the categorical construction works for substitutions.

Note that the presented program reflects the structure of the proof. To improve efficiency a number of optimizations are possible, e.g. one can use machine integers to represent variables and weakenings.

```
infixr -->;

datatype Ty = O | --> of Ty*Ty;

type Co = Ty list;

datatype Tm = var0 | varS of Tm | lam of Tm | app of Tm*Tm;

datatype Wk = w_id | w1 of Wk | w2 of Wk;

val w_pi = w1(w_id);

fun o_wk w_id   w  = w
  | o_wk (w1 w) w' = w1(o_wk w w')
  | o_wk (w2 w) w_id = w2(w)
  | o_wk (w2 w) (w1 w') = w1(o_wk w w')
  | o_wk (w2 w) (w2 w') = w2(o_wk w w');

fun wk_tm w_id M = M
  | wk_tm w (lam M) = lam (wk_tm (w2 w) M)
  | wk_tm w (app (M1,M2)) = app(wk_tm w M1,wk_tm w M2)
  | wk_tm (w2 w) (varS M) = wk_tm w M
  | wk_tm (w2 w) var0 = var0
  | wk_tm (w1 w) M    = varS (wk_tm w M);
```

```
datatype Vl = v_o of Tm | v_arr of Wk->Vl->Vl;

fun wk_vl w (v_o M) = v_o (wk_tm w M)
  | wk_vl w (v_arr f) = v_arr (fn w' => fn x => f (o_wk w w') x);

fun wk_vls w xs = map (wk_vl w) xs;

fun q O (v_o M) = M
  | q (S --> T) (v_arr f) = lam (q T (f w_pi (u S var0)))
and u O M = v_o M
  | u (S --> T) M =
      v_arr(fn w => fn x => (u T (app (wk_tm w M,q S x))));

fun eval var0     (x::xs) = x
  | eval (varS M) (x::xs) = eval M xs
  | eval (lam M) xs =
      v_arr (fn w => fn x => eval M (x::(wk_vls w xs)))
  | eval (app (M1,M2)) xs =
      case (eval M1 xs) of
          (v_arr f) => f w_id (eval M2 xs);

fun id [] = []
  | id (S::G) = (u S var0)::(wk_vls w_pi (id G));

fun nf G S M = q S (eval M (id G));
```

References

[Ber94] Ulrich Berger. Personal email to Thomas Streicher, April 1994.

[BS91] Ulrich Berger and Helmut Schwichtenberg. An inverse of the evaluation functional for typed λ-calculus. In *Proceedings of the Sixth Annual Symposium on Logic in Computer Science*, pages 202 – 211, 1991.

[CD93] Thierry Coquand and Peter Dybjer. Intuitionistic model constructions and normalization proofs. Preliminary Proceedings of the 1993 TYPES Workshop, Nijmegen (accepted for publication in *Mathematical Structures in Computer Science*), 1993.

[Coq91] Thierry Coquand. An algorithm for testing conversion in type theory. In *Logical Frameworks*. Cambridge University Press, 1991.

[Coq94] Catarina Coquand. From Semantics to Rules: a Machine Assisted Analysis. In Börger, Gurevich, and Meinke, editors, *CSL'93*, pages 91–105. Springer-Verlag, LNCS 832, 1994.

[Gha95] Neil Ghani. $\beta\eta$-equality for coproducts. In Mariangiola Dezani-Ciancaglini and Gordon Plotkin, editors, *Typed Lambda Calculi and Applications*, number 902 in LNCS, pages 171–185, 1995.

[HS95] Gérard Huet and Amokrane Saïbi. Constructive category theory. In Peter Dybjer and Randy Pollack, editors, *Informal proceedings of the joint CLICS-TYPES workshop on categories and type theory*, 1995.

[Kri65] Saul A. Kripke. Semantical analysis of intutionistic logic I. In J.N. Crossley and M.A.E. Dummett, editors, *Formal systems and recursive functions.* North Holland, 1965.

[LS86] Joachim Lambek and Phil Scott. *Introduction to Higher Order Categorical Logic.* Cambridge University Press, 1986.

[ML75] Per Martin-Löf. An Intuitionistic Theory of Types: Predicative Part. In H. E. Rose and J. C. Shepherdson, editors, *Logic Colloquium 1973*, pages 73–118, Amsterdam, 1975. North-Holland Publishing Company.

[MM91] John C. Mitchell and Eugenio Moggi. Kripke-style models for typed lambda calculus. *Annals of Pure and Applied Logic*, 51:99–124, 1991.

[NPS90] Bengt Nordström, Kent Petersson, and Jan M. Smith. *Programming in Martin-Löf's Type Theory. An Introduction.* Oxford University Press, 1990.

[TvD88] A. S. Troelstra and D. van Dalen. *Constructivism in Mathematics. An Introduction*, volume II. North-Holland, 1988.