The Groupoid Model Refutes Uniqueness of Identity Proofs

Martin Hofmann
Dept. of Computer Science
Univ. of Edinburgh
Edinburgh, Scotland

Thomas Streicher
Institut für Informatik
Ludwig-Maximilian-Universität
Munich, Germany

Abstract

We give a model of intensional Martin-Löf type theory based on groupoids and fibrations of groupoids in which identity types may contain two distinct elements which are not even propositionally equal. This shows that the principle of uniqueness of identity proofs is not derivable in the syntax.

1 Motivation and Main Result

When representing logic via the Curry-Howard paradigm of propositions-as-types one is forced to introduce a so-called identity type $\mathrm{Id}_A(a,b)$ consisting of all proof objects for the proposition "the objects a and b of type A are equal". In extensional type theory propositional equality $\mathrm{Id}_A(a,b)$ is identified with judgemental equality $a = b : A$ via the identity reflection rule, which allows to conclude judgemental equality from propositional equality, cf. [5]. Unfortunately, this identification leads to the non-confluence of the rewrite system determining the operational semantics of the term language [9]. Even worse, type-checking and thus proof-checking becomes undecidable.

Therefore — in order to keep the computational meaning of proofs — one has to restrict oneself to intensional constructive type theory where for any type A the identity type is introduced as an inductively defined family of types $x : A, y : A \vdash \mathrm{Id}_A(x,y)$ with constructor $x : A \vdash \mathrm{refl}_A(x) : \mathrm{Id}_A(x,x)$ and eliminator J whose meaning is given by the following rules

$$\frac{\Gamma, x : A, y : A, z : \mathrm{Id}_A(x,y) \vdash C(x,y,z) \qquad \Gamma, x : A \vdash d : C(x,x,\mathrm{refl}_A(x))}{\Gamma, x : A, y : A, z : \mathrm{Id}_A(x,y) \vdash J_{A,C}(d,x,y,z) : C(x,y,z) \qquad \Gamma, x : A \vdash J_{A,C}(d,x,x,refl_A(x)) = d : C(x,x,refl_A(x))}$$

Surprisingly enough, the J-eliminator is sufficient for constructing terms refl, symm, trans and subst inhabiting the types corresponding to the propositions expressing reflexivity, symmetry, transitivity and replacement. However, in a proof term the application of the replacement rule is always mentioned and cannot be eliminated in general. This means that in the situation $x,y : A$, $p : \mathrm{Id}_A(x,y)$, $b : B(x)$ we have $\mathrm{subst}_{A,B}(p,b) : B(y)$ but not $b : B(y)$ like in extensional type theory. For the definition of these combinators from J see e.g. [7].

Notice the absence of an η-rule for J which would in fact allow to derive the identity reflection rule, cf. [9], and therefore destroy the operational semantics of the term language because relative to certain nonempty contexts distinct variables would become judgementally equal, namely the variables x and y w.r.t the context $x : A, y : A, z : \mathrm{Id}_A(x,y)$.

A further manifestation of the *a priori* unexpected strength of the eliminator J is that one can derive proof terms for the types corresponding to the propositions expressing in terms of propositional equality the η-rules for various inductively defined types, cf. [7]. For example one can find an inhabitant of the type

$$\Pi p : \Sigma a : A.B(a).\mathrm{Id}_{\Sigma x:A.B(x)}(p\,,\,\langle p.1, p.2\rangle)$$

thus "surjective pairing" holds propositionally. Similarly, the type

$$\Pi n : \mathrm{N}.\mathrm{Id}_{\mathrm{N}}(n,0) + \Sigma n' : \mathrm{N}.\mathrm{Id}_{\mathrm{N}}(n,\mathrm{S}(n'))$$

is inhabited. The intuitive meaning of these propositional versions of η-rules is that "any object of an inductive type is propositionally equal to an object in constructor form".

The main result of this paper states that a corresponding property does not hold for the identity type itself. Indeed, below in Section 3 we will prove

Theorem 1 *There exists a type A in the empty context such that the type*

$$x : A\,,\, p,q : Id_A(x,x) \vdash Id_{Id_A(x,x)}(p, refl_A(x))$$

is not inhabited.

A similar principle for $\mathrm{Id}_A(x,y)$ where x and y are not assumed to be necessarily equal is

$$x : A, y : A, p, q : \mathrm{Id}_A(x,y) \vdash \mathrm{Id}_{\mathrm{Id}_A(x,y)}(p,q)$$

One can show using J that this latter type is inhabited iff the former type is. Henceforth we say that *the principle UIP (uniqueness of identity proofs) holds for some type* A if either of the above types is inhabited for that type. Similarly, we say that UIP holds in a model of type theory, if either of the above types receives nonempty interpretation in the model.

For example, the principle UIP is intuitively valid according to the naive set-theoretic semantics where $\mathrm{Id}_A(x,y)$ is empty if x and y are different objects of A and is a singleton set otherwise. The principle UIP has some practical relevance because it

1043-6871/94 $3.00 © 1994 IEEE

is equivalent to the derivability of $\mathrm{Id}_{B(x)}(y,z)$ from $\mathrm{Id}_{\Sigma x:A.B(x)}(\langle x,y\rangle,\langle x,z\rangle)$ as shown in [9].

It is known that UIP can be derived using pattern-matching, cf. [2]. Thus according to the result of this paper pattern-matching is a non-conservative extension of intensional constructive type theory. A syntactic proof of the non-derivability of UIP is not known and – if it exists – probably too complicated to be presented in a transparent way. Therefore, in this paper we use a semantic method to establish independence of UIP. Now almost all known models are extensional in the sense that the Identity Reflection Rule is valid with respect to arbitrary contexts and therefore UIP holds trivially. Even in the non-extensional models studied in [9] UIP always holds. (This is rather an advantage than a disadvantage of these models. The model under consideration in this paper we don't consider as an intended model it rather serves the purpose of providing the desired independence of UIP.) A syntactic argument shows that whenever UIP holds for *some* interpretation of the Id-type then UIP holds for all possible interpretations of the Id-type in the same model. Thus in order to disprove UIP we must look for a model which does *not admit any* interpretation of the extensional identity types. We will show that such a model can be obtained by interpreting types as *groupoids*.

2 The Groupoid Model

Although by lack of UIP not all proof object of identity types are propositionally equal nevertheless various equational laws do hold which induce a natural groupoid structure on every type A. That means that for appropriately typed p,q,r the following types are always provably inhabited

$\mathsf{Id}(\mathsf{trans}(p,\mathsf{trans}(q,r))\,,\,\mathsf{trans}(\mathsf{trans}(p,q),r))$	TRANS-ASSOC
$\mathsf{Id}(\mathsf{trans}(\mathsf{refl}(x),p)\,,\,p)$	NEUTR-L
$\mathsf{Id}(\mathsf{trans}(p,\mathsf{refl}(x))\,,\,p)$	NEUTR-R
$\mathsf{Id}(\mathsf{trans}(\mathsf{symm}(p),p)\,,\,\mathsf{refl}(x))$	SYMM-INV-L
$\mathsf{Id}(\mathsf{trans}(p,\mathsf{symm}(p))\,,\,\mathsf{refl}(x))$	SYMM-INV-R
$\mathsf{Id}(\mathsf{symm}(\mathsf{symm}(p)),p)$	INVOL

Thus we can show using J that the identity type Id_A endows every type A with a groupoid structure where the groupoid laws are satisfied up to propositional equality. This suggests to construct a model where types are interpreted as groupoids and where $\mathrm{Id}_A(x,y)$ intuitively does not mean equality but "being isomorphic", i.e. $\mathrm{Id}_A(x,y)$ will be interpreted as the discrete groupoid whose objects are the morphisms from the interpretation of x to the interpretation of y. In this model the interpretations of Id_A, refl, symm, and trans endow a type A with a groupoid structure satisfying the corresponding equational laws up to actual and not only up to propositional equality.

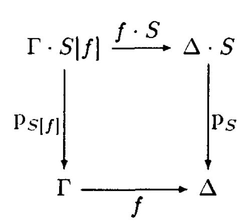
First we recall some basic definitions. For generalities about the theory of fibrations see the seminal [1](unfortunately unpublished).

Groupoids and fibrations of groupoids. A groupoid is a category where all morphisms are isomorphisms. A functor $p:\Delta\to\Gamma$ between groupoids Δ and Γ is a fibration iff for any $f:\gamma\to\gamma'$ in Γ and δ in the fibre over γ, i.e. $p(\delta)=\gamma$, there exists a morphism $\phi:\delta\to\delta'$ (for some δ' over γ') over f, i.e. $p(\phi)=f$. Equivalently, for every δ' above γ' there exists a morphism $\psi:\delta\to\delta'$ above f. A fibration of groupoids is *split* if there is a given choice of these morphisms compatible with the groupoid structure, i.e. the chosen morphism above the identity is the identity and the chosen morphism above a composition is the composition of the chosen morphisms.

If S is a covariant functor from groupoid Γ to the category of groupoids then the *Grothendieck construction* associates with S a split fibration $\mathrm{gr}(S):\mathrm{Gr}(S)\to\Gamma$. The groupoid $\mathrm{Gr}(S)$ has as underlying class of objects the pairs $\langle\gamma,s\rangle$ with $\gamma\in|\Gamma|$ and $s\in|S(\gamma)|$ and morphisms from $\langle\gamma,s\rangle$ to $\langle\gamma',s'\rangle$ are pairs $\langle f,\phi\rangle$ such that $f:\gamma\to\gamma'$ and $\phi:S(f)(s)\to s'$. Composition of morphisms is given by $\langle g,\psi\rangle\circ\langle f,\phi\rangle=\langle g\circ f,\psi\circ S(f)(\phi)\rangle$. The fibration $\mathrm{gr}(S)$ itself is the projection on the first component.

Categories with attributes. As an abstract frame of model we use the by now well-established notion of "category with attributes". Recall from [8, 6] that such a structure is given by

- a category Con with terminal object 1
- a presheaf $\mathrm{Fam}:\mathrm{Con}^{\mathrm{op}}\to\mathbf{Set}$ with morphism part written $\mathrm{Fam}(f)(S)=_{\mathsf{abbr}}S[f]$
- an operation p which to each $S\in\mathrm{Fam}(\Gamma)$ associates a Con-morphism $\mathrm{p}_S:\Gamma\cdot S\to\Gamma$ — the *canonical projection* of S
- An operation $\cdot$ which to each Con-morphism $f:\Gamma\to\Delta$ and $S\in\mathrm{Fam}(\Delta)$ associates a morphism $f\cdot S:\Gamma\cdot S[f]\to\Delta\cdot S$ such that [1]



is a pullback and the coherence conditions $\mathrm{id}_\Delta\cdot S=\mathrm{id}_{\Delta\cdot S[f]}$ and $(f\circ g)\cdot S=(f\cdot S)\circ(g\cdot S[f])$ for $g:\Theta\to\Gamma$ are satisfied.

Provided that suitable interpretations of base types and type constructors are given, a partial interpretation function can be defined by structural induction in such a way that every context is interpreted as a Con-object, every type is interpreted as an element of Fam at the interpretation of its context and finally terms are interpreted as *sections* (right inverses) of the canonical projections associated to their types. This

[1] This and the following diagrams have been typeset using Paul Taylor's diagram macros.

interpretation is sound in the sense that the interpretation of all derivable judgements is defined and that all equality judgements are validated w.r.t. the actual equality in the model.

We now turn to the definition of a particular category with attributes based on the category of groupoids. Let Con denote the category of all not necessarily small groupoids and functors between them. We put Fam := Func(−, Con), i.e. for $\Gamma \in |\mathrm{Con}|$ let $\mathrm{Fam}(\Gamma)$ be the class of functors from Γ viewed as a category to the category of groupoids, and if in addition $f : \Delta \to \Gamma$ then $S[f]$ is the functor $S \circ f : \Delta \to \mathrm{Con}$. The canonical projection $\mathrm{p}_S : \Gamma \cdot S \to \Gamma$ is the fibration of groupoids associated to S by the Grothendieck construction and finally $f \cdot S : \Delta \cdot S[f] \to \Gamma \cdot S$ is the second projection of the canonical pullback of p_S along f. Its effect on objects is $\Delta \cdot S[f] \ni (\delta, s) \mapsto (f(\delta), s) \in \Gamma \cdot S$.

Proposition 1 *The above data define a category with attributes.*

Now we will describe the interpretation of the type constructors.

2.1 Dependent Products and Sums of Families of Types

First we show that in groupoids and fibrations of groupoids dependent products and sums can be interpreted.

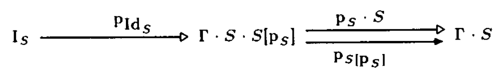
Proposition 2 *The groupoid model can be endowed with dependent products and sums of families.*

We sketch the construction for the case of dependent products. Let $\Gamma \in |\mathrm{Con}|$, $S \in \mathrm{Fam}(\Gamma)$ and $T \in \mathrm{Fam}(\Gamma \cdot S)$. First we need some auxiliary notation. Let $\mathrm{p}_\gamma : T_\gamma \to S(\gamma)$ be the fibration associated by the Grothendieck construction to the family $S(\gamma) \to \mathrm{Con}$ obtained by restricting T along the inclusion of $S(\gamma)$ into $\Gamma \cdot S$. For $a : \gamma \to \gamma'$ let $T_a : T_\gamma \to T_{\gamma'}$ denote the functor defined by $T_a(s,t) = (S(a)(s), T(a, \mathrm{id}_{S(a)(s)})(t)$ and $T_a(b,c) = (S(a)(b), T(a, \mathrm{id}_{S(a)(s)})(c)$.

Then $\Pi(S,T) \in \mathrm{Fam}(\Gamma)$ is given as follows. For $\gamma \in |\Gamma|$ we define $\Pi(S,T)(\gamma)$ as the groupoid of all functors $M : S(\gamma) \to T_\gamma$ such that $\mathrm{p}_\gamma \circ M = \mathrm{id}_{(}\gamma)$. Whenever $a : \gamma \to \gamma'$ then $\Pi(S,T)(a) : \Pi(S,T)(\gamma) \to \Pi(S,T)(\gamma')$ is defined by putting $\Pi(S,T)(a)(M) = T_a \circ M \circ S(a^{-1})$ and the morphism part is defined analogously.

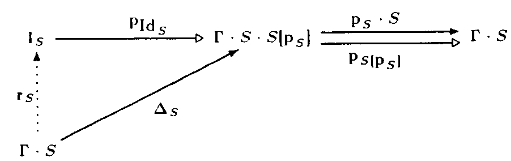
2.2 The Interpretation of the Identity Type

Let $\Gamma \in |\mathrm{Con}|$ and $S \in \mathrm{Fam}(\Gamma)$. Then $\mathrm{Id}_S \in \mathrm{Fam}(\Gamma \cdot S \cdot S[\mathrm{p}_S])$ is defined as follows. For $\gamma \in |\Gamma|$ and $s, t \in |S(\gamma)|$ let $\mathrm{Id}_S(\gamma, s, t)$ be the discrete groupoid with $S(\gamma)(s,t)$ as underlying set of objects. The unique element of a homset $\mathrm{Id}_S(\gamma)(s,s)$ is denoted $\star$. If furthermore $a : \gamma \to \gamma'$, $b : S(a)(s) \to s'$ in $S(\gamma')$ and $c : S(a)(t) \to t'$ in $S(\gamma')$ (i.e. the triple (a,b,c) is a morphism from (γ, s, t) to (γ', s', t') in $\Gamma \cdot S \cdot S[\mathrm{p}_S]$) then for $i \in \mathrm{Id}_S(\gamma, s, t)$ we put $\mathrm{Id}_S(a,b,c)(i) = c \circ S(a)(i) \circ b^{-1}$. The action on morphisms is defined analogously. Notice that the total object $\mathrm{I}_S := \Gamma \cdot S \cdot S[\mathrm{p}_S] \cdot \mathrm{Id}_S$ of this family is the fibrewise arrow category of S or equivalently the *inserter* of the two projections from $\Gamma \cdot S \cdot S[\mathrm{p}_S]$ to $\Gamma \cdot S$.

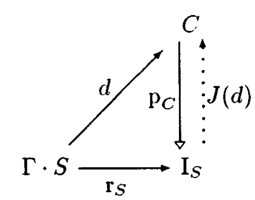


Recall from [3] that the inserter is a 2-categorical generalisation of the concept of equalizer in which equality is replaced by natural transformation.

Reflexivity. In order to interpret the constructor refl we define a section of the canonical projection of $\mathrm{Id}_S[\Delta_S]$ where $\Delta_S : \Gamma \cdot S \to \Gamma \cdot S \cdot S[\mathrm{p}_S]$ is the fibrewise diagonal. We define $\mathrm{refl}_S(\gamma)(t) := (\gamma, t, t, id_t)$ which extends uniquely to a morphism. Using the presentation of Id as an inserter we get refl or rather r_S, its composition with $\Delta_S \cdot \mathrm{Id}_S$, as the unique fill-in w.r.t the identity 2-cell in



Identity elimination. Finally in order to interpret the eliminator J assume that $C \in \mathrm{Fam}(\mathrm{I}_S)$ and that d is a section of the canonical projection of the family $C[\Delta_S \cdot \mathrm{Id}_S][\mathrm{refl}_S] \in \mathrm{Fam}(\Gamma \cdot S)$. The we define the section $J(d)$ of p_C by putting $J(d)(\gamma, s, t, p) = C(\mathrm{id}_\gamma, \mathrm{id}_s, p, \star)(d(\gamma, s))$. Notice here that $(\mathrm{id}_\gamma, \mathrm{id}_s, p, \star)$ is a morphism in I_S from $(\gamma, s, s, \mathrm{id}_s)$ — an object in the domain of d, to (γ, s, s', p) — a most general object of I_S. The morphism part is defined similarly. Notice that although this interpretation is somewhat arbitrary (it depends on the choice of the morphism parts of the involved presheaves S and C) the whole situation is nevertheless stable under reindexing of Γ. More categorically, the interpretation of J gives us a canonical (stable under pullback) factorisation of morphisms $d : \Gamma \cdot S \to C$ with $\mathrm{p}_C \circ d = \mathrm{r}_S$ through r_S.



such that moreover $J(d)$ is a section of p_C. It is possible to derive this property from the characterisation of Id as an inserter alone. It is of course important and a distinctive feature of the groupoid model that this inserter arises as a canonical projection (is a split fibration).

Intensionality of the groupoid model. This interpretation of the identity type is intensional in the sense that in the model the identity reflection rule is not validated as in the model propositional equality only means isomorphism, but not actual equality. For the same reason unlike in the intensional models studied in [9] (and in fact in the term model) the equality reflection rule is not even valid in the empty context (i.e. for global elements). But it is valid for all types which are interpreted as discrete groupoids (i.e. especially for all identity types and types which are syntactically definable without a universe). Furthermore in the groupoid model the type

$$f, g : A \to B \vdash (\Pi a : A.\mathrm{Id}_B(f\,a, g\,a)) \to \mathrm{Id}_{A\to B}(f, g)$$

is inhabited (by the identity!) which is a rather extensional feature.

2.3 The Other Type Constructors

The type N of natural numbers gets interpreted as the discrete groupoid whose underlying set of objects is the set of natural numbers. The (first) universe U gets interpreted as the non-small discrete groupoid whose underlying set is the set of all small groupoids. The generic family gets interpreted as the embedding functor from U to the category of all small groupoids.

We conjecture that all other type formers considered e.g. in [7] can be interpreted as well.

3 The Independence Result

Suppose Γ is a groupoid with an object γ with a nontrivial endomorphism $a \in \Gamma(\gamma,\gamma)$. Now if UIP could be derived uniformly using J its interpretation in the model would give rise to an object in $\mathrm{Id}_{\mathrm{Id}_\Gamma(\gamma,\gamma)}(\mathrm{refl}_\Gamma(\gamma), a)$. But since $\mathrm{Id}_\Gamma(\gamma,\gamma)$ is discrete this entails that $a = \mathrm{refl}_\Gamma(\gamma)$ contradicting the assumption. This does not contradict UIP for certain specific types as e.g. N, List(N) etc. which in our model get interpreted as discrete groupoids for which UIP trivially holds as any homset contains at most one element. Indeed, for the latter two types UIP is actually syntactically derivable in the presence of a universe. It is still an open problem whether UIP is derivable for $\mathrm{N} \to \mathrm{N}$ which in our model gets interpreted by a discrete groupoid, too, and therefore UIP does hold for $\mathrm{N} \to \mathrm{N}$.

A syntactically definable type for which UIP fails is $\Sigma X : U.\mathrm{El}(X)$ as there is a small groupoid containing an object admitting two different endomorphisms. Thus we have proved Theorem 1. Again it is an open problem whether there exists a type definable without universe for which UIP fails. Our model cannot solve this problem since the interpretations of all such types are discrete.

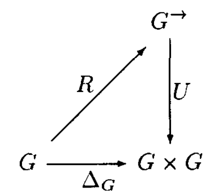
3.1 Another identity elimination rule

In [9] it has been shown that the validity of UIP is equivalent to the existence of a further eliminator K for Id-types (independently found by Th. Altenkirch) whose meaning is given by the following rules:

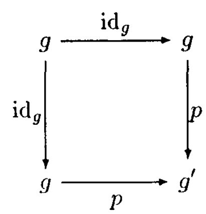
$$\frac{\begin{array}{c}\Gamma, x : A, z : \mathrm{Id}_A(x,x) \vdash C(x,z) \\ \Gamma, x : A \vdash d : C(x, \mathrm{refl}_A(x))\end{array}}{\begin{array}{c}\Gamma, x : A, z : \mathrm{Id}_A(x,x) \vdash K_{A,C}(d,x,z) : C(x,z) \\ \Gamma, x : A \vdash K_{A,C}(d,x,\mathrm{refl}_A(x)) = d : C(x,\mathrm{refl}_A(x))\end{array}}$$

Intuitively, K is the eliminator for the inductive family $x : A \vdash \mathrm{Id}_A(x,x)$ where the inductive structure is inherited from the family $x : A, y : A \vdash \mathrm{Id}_A(x,y)$. We will now give a direct proof of the fact that the eliminator K cannot be interpreted in the groupoid model. (There is an indirect proof because employing K we can derive UIP!). In this proof we are a bit less formal and in this way try to highlight the main intuitive idea behind the identity type in the groupoid model.

We use the syntax informally to denote objects in the groupoid model. Moreover, we only consider Id for a groupoid (viewed as an element of Fam(1)) and not for arbitrary families. For any groupoid G the fibration associated to the family $x, y : G \vdash \mathrm{Id}_G(x,y)$ is given by the functor $U = \langle \partial_0, \partial_1 \rangle : G^{\to} \to G \times G$ where $G^{\to}$ is the arrow category of G and ∂_0, ∂_1 are the domain and codomain functors. The constructor refl_G gives a functor R making



commute. R sends $g \in |G|$ to id_g. If $q : C \to G^{\to}$ is a (split) fibration associated to a family as in the premise of J then from a morphism $d : G \to C$ such that $q \circ d = R$ we get a section $J(d)$ of q. Its construction hinges on the fact that in $G^{\to}$ every object $p : g \to g'$ is connected to an identity morphism as follows.



As shown above in Section 2.2 we get $J(d)$ from d by reindexing (in q) along this morphism.

Now in the premise to K the situation is entirely different. The diagonalised family $g : G \vdash \mathrm{Id}_G(g,g)$ is the pullback of $U : G^{\to} \to G$ along the diagonal Δ_G. More explicitly it is the fibration $V : G^{\mathrm{endo}} \to G$ where G^{endo} is the category with objects (g,p) where $g \in |G|$ and $p : g \to g$ and a morphism from (g,p) to (g',p') is a G-morphism $f : g \to g'$ such that $p' \circ f = f \circ p$ and V is the first projection. Unlike in the category

$G^{\rightarrow}$ in G^{endo} the identities are *isolated*. If there is a morphism f from (g, id_g) to (g', p)

$$\begin{array}{ccc} g & \xrightarrow{f} & g' \\ {\scriptstyle \mathrm{id}_g}\downarrow & & \downarrow{\scriptstyle p} \\ g & \xrightarrow[f]{} & g' \end{array}$$

then $p = \mathrm{id}_{g'}$ as $p \circ f = \mathrm{id}_{g'} \circ f$. Now if we had an interpretation of the eliminator K in the groupoid model then for any split fibration $q : C \to G^{\text{endo}}$ and functor $d : G \to C$ with $q \circ d = r_G$ (here r_G is meant to have codomain G^{endo} !) we would get a section $K(d)$ of q. But now since the identities are isolated in G^{endo}, the morphism r_G is such a fibration itself! So we get a section of r_G which is a contradiction in case G has an object with a nontrivial endomorphism.

4 Conclusion and Open Problems

In this paper we have shown that the current formulation of intensional constructive type theory is incomplete w.r.t. the intuitive understanding of identity types. Namely, it is possible to construct a model where not all proof objects of an identity type are necessarily propositionally equal (that - relative to nonempty contexts - they are not always judgementally equal is a characteristic feature of intensional type theory anyway). For this purpose we have constructed a model where types are interpreted as groupoids, functions as morphisms of groupoids and propositional equality as "being isomorphic" where the proof objects in type $\mathrm{Id}_A(t, s)$ are exactly the isomorphisms of the interpretations of t and s, respectively. As the basic types which can be defined without a universe get interpreted by discrete groupoids in our model for them the principle UIP (= Unicity of Identity Proofs) holds nevertheless. But our result shows that for each such type — in order to prove UIP for it — requires a new idea, i.e. it cannot be done uniformly.

Supposedly, UIP is not provable for the type $\mathrm{N} \to \mathrm{N}$. It might be the case that this can be proved in a semantical way by synthesizing the groupoid model of this paper with the modified realizability model described in Chapter 3 of [9].

Our work suggests that the intensional identity type can be used as an "internal language" for groupoid-like structures. In particular it seems worthwhile to investigate whether the intensional identity type facilitates the monstruous bookkeeping which has to be employed when dealing with coherent isomorphism instead of actual equality. This idea ties in with the approach of [4] where, however, the rules for the intensional identity type are not used.

References

[1] Jean Benabou. Des Categories Fibrees. Lectures notes by J. Roisin on a course given 1980 in Louvain-la-Neuve.

[2] Thierry Coquand. Pattern Matching with Dependent Types. In *Workshop on Logical Frameworks, Båstad*, 1992. Preliminary Proceedings.

[3] Gregory Maxwell Kelly and Ross Street. Review of the elements of 2-categories. In *Proceedings of the Category Seminar, Sydney*, pages 75–103. Springer, LNM 420, 1972/73.

[4] François Lamarche. A Proposal about Foundations I. Manuscript.

[5] Per Martin-Löf. *Intuitionistic Type Theory.* Bibliopolis-Napoli, 1984.

[6] Eugenio Moggi. A category-theoretic account of program modules. *Math Struct. in Comp. Sci.*, 1(1):103–139, 1991.

[7] B. Nordström, K. Petersson, and J. M. Smith. *Programming in Martin-Löfs Type Theory, An Introduction.* Clarendon Press, Oxford, 1990.

[8] Andrew Pitts. Categorical logic. In *Handbook of Theoretical Computer Science.* Elsevier Science, 199? to appear.

[9] Thomas Streicher. *Investigations into Intensional Type Theory.* Habilitationsschrift, LMU München, 1993.