Equality and dependent type theory

Thierry Coquand

24th AILA meeting, Bologna, February 2-4

Equality and dependent type theory

This talk: comments on V. Voevodsky *Univalent Foundation for Mathematics*

A refinement of the *propositions-as-types* or *Curry-Howard* interpretation

type = proposition = homotopy type of a space

Foundations of Mathematics

In the same year 1908

set theory, Zermelo *Investigations in the foundation of set theory*

type theory, Russell *Mathematical logic as based on the theory of types*

How do these two foundations compare w.r.t. equality?

Equality in set theory

Axiom of extensionality: two sets that have the same elements are equal

Axiom I in Zermelo's 1908 article

Equality in type theory

First edition of *Principia Mathematica* (1910): no axiom of extensionality, but axiom of reducibility (propositions form a type, and we can quantify over propositions, also known as *impredicativity*)

Second edition (1925): under the influence of Wittgenstein, Russell introduces the principle of extensionality

a function of propositions is always a truth function, and a function occurs only in a proposition through its values

and sees this as a (partial) replacement of the axiom of reducibility

Equality in type theory

A function can only appear in a matrix though its values

"This assumption is fundamental in the following theory. It has its difficulties, but for the moment, we ignore them. It takes the place (not quite adequatly) of the axiom of reducibility"

Church's formulation of type theory

Simplification of Russell's theory of types

A type of proposition o, a type of individuals and function type $A \to B$

For instance $o \to o$ is the type of the operation of negation

We have the usual connectives on propositions

$p \to q : o$ for the implication if $p\ q : o$

quantifiers at *any* type $\forall x : A.\varphi : \ o$ if $\varphi : o\ [x : A]$

Church's formulation

Uses λ-calculus to represent terms (implicit in *Principia Mathematica*)

If $f : A \rightarrow B$ and $a : A$ then $f\ a : B$ the application of the function f to the argument a

If $t : B\ [x : A]$ then $\lambda x.t : A \rightarrow B$

The terms of type o are the propositions

Usual connectives and (classical) logical rules

Equality in Church's formulation

We can *define* an equality (Leibniz equality) $\mathsf{Id}_A\ a_0\ a_1$ as

$\forall P : A \rightarrow o.\ \ P(a_0) \rightarrow P(a_1)$

This definition is *impredicative*

One can show that this is a reflexive, symmetric and transitive relation

The *axiom of extensionality* has then two forms

on propositions: $(p \leftrightarrow q) \rightarrow \mathsf{Id}_o\ p\ q$

on functions: $(\forall x : A.\ \mathsf{Id}_B\ (f\ x)\ (g\ x)) \rightarrow \mathsf{Id}_{A\rightarrow B}\ f\ g$

Dependent Type Theory

Curry-Howard, N. de Bruijn, D. Scott, P. Martin-Löf

Add to simple type theory the notion of $\mathit{dependent\ type}$ $B(x)$ type for $x : A$

$\prod_{x:A} B(x)$ type of functions/sections f with $f\ a : B(a)$ if $a : A$

$\sum_{x:A} B(x)$ type of pairs a, b with $a : A$ and $b : B(a)$

Natural set theoretic interpretation

Proposition as Types

If $B(x) = B$ does not depend on $x : A$

$\prod_{x:A} B(x)$ is written $A \to B$ represents both function types and implication

$\sum_{x:A} B(x)$ is written $A \times B$ represents both cartesian product and conjunction

Proposition as Types

$$\prod_{x:A} B(x)$$ represents

-universal quantification and

-the set of sections of the family $B(x)$

Proposition as Types

$$\sum_{x:A} B(x) \text{ represents}$$

-the fiber space over A defined by the family $B(x)$ and

-the set $\{x : A \mid B(x)\}$ and

-existential quantification $(\exists x : A)B(x)$

Universe

Martin-Löf (1972) introduces the notion of universe U, type of "small" types

U can be thought of both as a type of types and as a type of propositions

$Predicative$ system

$\sum_{X:U} X \times (X \to X)$ or $\prod_{X:U} (X \to X)$ are large types and not of type U

$$\sum_{X:U} X \times (X \to X)$$

type of all structures with one constant and one unary operation

Some Notations

$A \to B \to C$ for $A \to (B \to C)$

$\lambda x\ y\ z.t$ for $\lambda x \lambda y \lambda z.t$

$\prod_{x_0\ x_1 : A} B(x_0, x_1)$ for $\prod_{x_0 : A} \prod_{x_1 : A} B(x_0, x_1)$

Dependent Type Theory

To summarize: extension of Gödel's system T with

$\prod_{x:A} B(x)$ and $\sum_{x:A} B(x)$

A type of small types U (closed under products and sums)

$N_0,\ N_1,\ N_2,\ N : U$

Terms: λ-terms extended with constants $0 : N$ and $x+1 : N\ [x : N]$ and

$\mathsf{natrec} : P(0) \rightarrow (\prod_{x:N} P(x) \rightarrow P(x+1)) \rightarrow \prod_{x:N} P(x)$

$\mathsf{natrec}\ a\ f\ 0 = a$ and $\mathsf{natrec}\ a\ f\ (n+1) = f\ n\ (\mathsf{natrec}\ a\ f\ n)$

Dependent Type Theory

Uniform foundation for logic and type theory: True $=$ Provable $=$ Inhabited

(In Church's type theory, one needs to add logical rules to the type structure)

For instance

$$\prod_{A\ B:U} (A \to B \to A)$$

is true because it is inhabited by $\lambda A\ B\ x\ y.\ x$

$$A : U,\ B : U \vdash \lambda x\ y.\ x : A \to B \to A$$

$$A : U,\ B : U,\ x : A,\ y : B \vdash x : A$$

Propositions as Types

Not so intuitive to consider the type N of natural number to be a proposition

A satisfactory answer will be provided by an analysis of equality in type theory

Equality in Dependent Type Theory

We follow an *axiomatic* approach: what should be the property of equality?

We should have a *type* of equality proofs $\mathsf{Id}_A\ a_0\ a_1$ if A type and $a_0\ a_1 : A$

We write $\alpha, \beta, \ldots$ equality proofs

Some axioms

$1_a : \mathsf{Id}_A\ a\ a$ if $a : A$

$(\cdot) : B(a_0) \to \mathsf{Id}_A\ a_0\ a_1 \to B(a_1)$ given $B(x)$ type over $x : A$

We have $b \cdot \alpha : B(a_1)$ if $b : B(a_0)$ and $\alpha : \mathsf{Id}_A\ a_0\ a_1$

Equality as Path

We think of a type A as a $space$

A proof $\alpha : \mathsf{Id}_A\ a_0\ a_1$ is thought of as a $path$ between a_0 and a_1

The operation $b \cdot \alpha : B(a_1)$ for $b : B(a_0)$ corresponds then to the *path lifting property*

(For a covering space, this lifting property provides a bijection between two fibers of two connected points)

We expect to have $\mathsf{Id}_{B(a_0)}\ (b \cdot 1_{a_0})\ b$

Equality as Path

3 axioms so far

$1_a : \mathsf{Id}_A\ a\ a$ if $a : A$

$(\cdot) : B(a_0) \rightarrow \mathsf{Id}_A\ a_0\ a_1 \rightarrow B(a_1)$

$\mathsf{ax}_3 : \mathsf{Id}_{B(a_0)}\ (b \cdot 1_{a_0})\ b$

Contractible Spaces

If A is a type we define a new type $\mathsf{iscontr}\ A$ to be $\sum_{a:A} \prod_{x:A} \mathsf{Id}_A\ a\ x$

This means that A has *exactly one element*

In term of space, A is *contractible*

A further axiom

(J.P.Serre) *when I was working on homotopy groups (around 1950), I convinced myself that, for a space X, there should exist a fibre space E, with base X, which is contractible; such a space would allow me (using Leray's methods) to do lots of computations on homotopy groups... But how to find it? It took me several weeks (a very long time, at the age I was then) to realize that the space of "paths" on X had all the necessary properties-if only I dared call it a "fiber space". This was the starting point of the loop space method in algebraic topology.*

A further axiom

Given a point a in X, J.P. Serre was considering the space E of paths α from a to another point x of A, with the map $E \to A,\ \alpha \longmapsto x$

E is contractible, and we have a contractible fibre space E with base X

In type theory, this translates to

For $a : X$, the type $E = \sum_{x:X} \mathsf{Id}_A\ a\ x$ should be contractible

Any element $(x, \alpha) : E$ is equal to $(a, 1_a)$

Equality as Path

4 axioms

$1_a : \mathsf{Id}_A\ a\ a$ if $a : A$

$(\cdot) : B(a_0) \to \mathsf{Id}_A\ a_0\ a_1 \to B(a_1)$

$\mathsf{ax}_3 : \mathsf{Id}_{B(a_0)}\ (b \cdot 1_{a_0})\ b$

$\mathsf{ax}_4 : \mathsf{iscontr}\ (\sum_{x:A} \mathsf{Id}_A\ a\ x)$

Equivalent formulation

introduction rule $1_a : \mathsf{Id}_A\ a\ a$

elimination rule: given $C(x,\alpha)$ for $x : A$ and $\alpha : \mathsf{Id}_A\ a\ x$ then we have

$$\mathsf{elim} : C(a,1_a) \to \prod_{x:A}\ \prod_{\alpha:\mathsf{Id}_A\ a\ x} C(x,\alpha)$$

(C. Paulin's formulation of equality in type theory)

"computation" rule: $\mathsf{Id}_{C(a,1_a)}\ (\mathsf{elim}\ c\ a\ 1_a)\ c$ for any $c : C(a,1_a)$

Dependent type version of $\mathsf{Id}_A\ a\ x \to P(a) \to P(x)$

Equivalent formulation

introduction rule $1_a : \mathsf{Id}_A\ a\ a$

elimination rule: given $C(x_0, x_1, \alpha)$ for $x_0\ x_1 : A$ and $\alpha : \mathsf{Id}_A\ x_0\ x_1$ we have

$$\mathsf{J} : (\prod_{x:A} C(x, x, 1_x)) \rightarrow \prod_{x_0\ x_1 : A}\ \prod_{\alpha : \mathsf{Id}_A\ x_0\ x_1} C(x_0, x_1, \alpha)$$

"computation" rule: $\mathsf{Id}_{C(x,x,1_x)}\ (\mathsf{J}\ d\ x\ x\ 1_x)\ (d\ x)$ for any $d : \prod_{x:A} C(x, x, 1_x)$

This is P. Martin-Löf's formulation of equality in type theory

It expresses in type theory that Id_A is the least reflexive relation on A

Consequences of these axioms

All these different formulations are equivalent axiom systems

Given these axioms any type has automatically a *groupoid structure*

Proofs-as-programs version of the fact that equality is symmetric and transitive

Any function $f : A \to B$ defines a functor

Hofmann-Streicher 1992

Equality as Path

Most topological intuitions have a direct formal expression in type theory, e.g.

for any type X and $a : X$ the type $\pi_1(X, a) = \mathsf{Id}_X\ a\ a$ has a group structure

$\pi_2(X, a) = \pi_1(\mathsf{Id}_X\ a\ a, 1_a), \ldots$ and we have

Proposition: $\pi_n(X, a)$ *is commutative for* $n \geqslant 2$

More generally, whenever we have a type X with a binary operation and an element $e : X$ which is both a left and right unit for this operation then the group $\pi_1(X, e) = \mathsf{Id}_X\ e\ e$ is commutative

Axiom of extensionality

The usual formulation of this axiom is, with $F = \prod_{x:A} B(x)$

$$(\prod_{x:A} \mathsf{Id}_{B(x)}\ (f\ x)\ (g\ x))) \rightarrow \mathsf{Id}_F\ f\ g$$

(V. Voevodsky) This is equivalent to

A product of contractible types is contractible

$$(\prod_{x:A} \mathsf{iscontr}\ (B(x))) \quad \rightarrow \mathsf{iscontr}\ (\prod_{x:A} B(x))$$

Equality as Path

5 axioms

$1_a : \mathsf{Id}_A\ a\ a$ if $a : A$

$(\cdot) : B(a_0) \to \mathsf{Id}_A\ a_0\ a_1 \to B(a_1)$

$\mathsf{ax}_3 : \mathsf{Id}_{B(a_0)}\ (b \cdot 1_{a_0})\ b$

$$\mathsf{ax}_4 : \mathsf{iscontr}\ (\sum_{x:A} \mathsf{Id}_A\ a\ x)$$

$$\mathsf{ax}_5 : (\prod_{x:A} \mathsf{iscontr}\ (B(x))) \to \mathsf{iscontr}\ (\prod_{x:A} B(x))$$

Stratification of types

A is of h-level 0 iff A is contractible

A is of h-level 1 iff $\mathsf{Id}_A\ a_0\ a_1$ is contractible for any $a_0\ a_1 : A$

A is a $proposition$ iff A is of h-level 1

A is of h-level 2 iff $\mathsf{Id}_A\ a_0\ a_1$ is a proposition for any $a_0\ a_1 : A$

A is a set iff A is of h-level 2

$\ldots$

Stratification of types

These definitions can be internalised in type theory

$$\text{isprop } A = \prod_{x_0\ x_1 : A} \text{iscontr } (\text{Id}_A\ x_0\ x_1)$$

$$\text{isset } A = \prod_{x_0\ x_1 : A} \text{isprop } (\text{Id}_A\ x_0\ x_1)$$

There is no "global" type of all propositions or of all sets

What "matters" is not the "size" of the type, but the complexity of its equality

Extensionality and impredicativity

The extensionality axiom implies

-a product of propositions is always a proposition

$$\prod_{x:A} \text{isprop}\ (B(x)) \quad \rightarrow \text{isprop}\ (\prod_{x:A} B(x))$$

-a product of sets is always a set

$$\prod_{x:A} \text{isset}\ (B(x)) \quad \rightarrow \text{isset}\ (\prod_{x:A} B(x))$$

The first implication confirms Russell's remark that the principle of extensionality can replace the axiom of reducibility

Propositions

If we have isprop $(B(x))$ for all $x : A$ then the canonical projection

$$(\sum_{x:A} B(x)) \rightarrow \quad A$$

is a mono, and we can think of $\sum_{x:A} B(x)$ as the *subset* of elements in A satisfying the property $B(x)$

Unique Existence

$\mathsf{iscontr}(\sum_{x:A} B(x))$ a generalisation of unique existence $\exists! x : A.B(x)$

If $B(x)$ is a proposition, $\mathsf{iscontr}(\sum_{x:A} B(x))$ reduces to unique existence on x

More refined in general than to state that only one element in A satisfies $B(x)$

We always have $\mathsf{iscontr}(\sum_{x:A} \mathsf{Id}_A\ a\ x)$ but $\mathsf{Id}_A\ a\ x$ may not be a proposition

Hedberg's Theorem

Define $\mathsf{isdec}\ A$ to be $\prod_{x_0\ x_1 : A} \mathsf{Id}_A\ x_0\ x_1 + \neg\ (\mathsf{Id}_A\ x_0\ x_1)$

$\neg\ C$ denotes $C \rightarrow N_0$, where N_0 is the empty type

M. Hedberg noticed (1995) that we have

$\mathsf{isdec}\ A \rightarrow \mathsf{isset}\ A$

In particular N the type of natural numbers is decidable

So N is a set but it is not a $proposition$ (since $\neg\ (\mathsf{Id}_N\ 0\ 1)$ is inhabited)

Other properties

$\mathsf{isprop}\ N_0, \quad \mathsf{iscontr}\ N_1, \quad \mathsf{isset}\ N_2$

$\neg\ A \rightarrow \mathsf{isprop}\ A$

$\mathsf{isprop}\ (\mathsf{iscontr}\ A)$ for all type A

$\mathsf{isprop}\ (\mathsf{isprop}\ A)$ for all type A

$\mathsf{isprop}\ (\mathsf{isset}\ A)$ for all type A

$\mathsf{isprop}\ A$ iff $\prod_{x_0\ x_1 : A} \mathsf{iscontr}(\mathsf{Id}_A\ x_0\ x_1)$ iff $\prod_{x_0\ x_1 : A} \mathsf{Id}_A\ x_0\ x_1$

Axiom of extensionality

In Church's type theory $(p \leftrightarrow q) \rightarrow \mathsf{Id}_o\ p\ q$

What about adding as an axiom $(X \leftrightarrow Y) \rightarrow \mathsf{Id}_U\ X\ Y$?

S. Berardi noticed that this is contradictory (with dependent type theory):

If X inhabited X is logically equivalent to $X \rightarrow X$

We would have $\mathsf{Id}_U\ X\ (X \rightarrow X)$ and then X and $X \rightarrow X$ are isomorphic

X model of λ-calculus, hence any map on X has a fixed-point

and we get a contradiction if $X = N$ or $X = N_2$

Axiom of extensionality

So we need a more subtle formulation

Define $\mathsf{Isom}\ X\ Y$ to be

$$\sum_{f:X\to Y}\ \sum_{g:Y\to X}\ (\prod_{x:X} \mathsf{Id}_X\ (g\ (f\ x))\ x) \times (\prod_{y:Y} \mathsf{Id}_Y\ (f\ (g\ y))\ y)$$

Extensionality axiom for small types (Hofmann-Streicher 1996)

$$\mathsf{Isom}\ X\ Y\ \ \to\ \ \mathsf{Id}_U\ X\ Y$$

Other properties

A consequence of this axiom is

$\neg(\text{isset } U)$

Indeed, $\text{Id}_U\ N_2\ N_2$ has two distinct elements

We have

If $\text{isset } A$ and $\prod_{x:A} \text{isset } (B(x))$ then $\text{isset } (\sum_{x:A} B(x))$

$\text{isset } A$ is not connected to the size of A but with the complexity of the equality on A

Equality as Path

6 axioms

$1_a : \mathsf{Id}_A\ a\ a$ if $a : A$

$(\cdot) : B(a_0) \to \mathsf{Id}_A\ a_0\ a_1 \to B(a_1)$

$\mathsf{ax}_3 : \mathsf{Id}_{B(a_0)}\ (b \cdot 1_{a_0})\ b$

$\mathsf{ax}_4 : \mathsf{iscontr}\ (\sum_{x:A} \mathsf{Id}_A\ a\ x)$

$\mathsf{ax}_5 : (\prod_{x:A} \mathsf{iscontr}\ (B(x))) \to \mathsf{iscontr}\ (\prod_{x:A} B(x))$

$\mathsf{ax}_6 : \mathsf{Isom}\ X\ Y \to \mathsf{Id}_U\ X\ Y$

Univalence Axiom

For $f : Y \to X$ and $x_0 : X$, the *fiber* of f above x_0 is

$$f^{-1}(x_0) =_{def} \sum_{y:Y} \mathsf{Id}_X\ x_0\ (f\ y)$$

$\sum_{x:X} f^{-1}(x) = \sum_{x:X} \sum_{y:Y} \mathsf{Id}_X\ x\ (f\ y)$ is the graph of f

Any map $f : Y \to X$ *is isomorphic to a fibration* $\sum_{x:X} f^{-1}(x) \to X$

Univalence Axiom

We define what should be a "path" between two types X and Y

If $f : X \rightarrow Y$ we define when f is a *weak equivalence*

$$\text{isweq } f \ =_{def} \ \prod_{y:Y} \text{iscontr } (f^{-1}(y))$$

Theorem: *To be a weak equivalence is always a proposition, i.e.* $\text{isprop } (\text{isweq } f)$

We define $\text{Weq } X \ Y$ to be $\sum_{f:X \rightarrow Y} \text{isweq } f$

Univalence Axiom

Let isiso f be

$$\sum_{g:Y\to X} (\prod_{x:X} \mathsf{Id}_X\ (g\ (f\ x))\ x) \times (\prod_{y:Y} \mathsf{Id}_Y\ (f\ (g\ y))\ y)$$

isiso $f \leftrightarrow$ isweq f

However isweq f is a always a proposition while

isiso f may not be a proposition in general

Univalence Axiom

Warning! Weak equivalence is $stronger$ than logical equivalence, e.g.

$$\prod_{x:A} \sum_{y:B} R(x,y) \text{ and } \sum_{f:A\to B} \prod_{x:A} R(x, f\ x)$$

are weakly equivalent, since they are isomorphic

This is more precise than only to state logical equivalence

Univalence Axiom

Clearly we have Weq X X, because the identity map is a weak equivalence

Hence we have a map

$\mathsf{Id}_U\ X\ Y \rightarrow \mathsf{Weq}\ X\ Y$

The *Univalence Axiom* states that this map is a weak equivalence

V. Voevodsky has shown that this implies functional extensionality

This axiom does not hold for the set-theoretic interpretation of type theory

Invariance under isomorphisms

We get a formalism where two *isomorphic* mathematical structures are *equal*

For instance on the type $S = \sum_{X:U} X \times (X \to X)$ we have

$\mathsf{Id}_S\ (X, a, f)\ (Y, b, g)$ iff

the structures (X, a, f) and (Y, b, g) are isomorphic

As we saw, this property does not hold for set theory

Is this theory *consistent*?

Equality as Path

6 axioms

$1_a : \mathsf{Id}_A\ a\ a$ if $a : A$

$(\cdot) : B(a_0) \to \mathsf{Id}_A\ a_0\ a_1 \to B(a_1)$

$\mathsf{ax}_3 : \mathsf{Id}_{B(a_0)}\ (b \cdot 1_{a_0})\ b$

$\mathsf{ax}_4 : \mathsf{iscontr}\ (\sum_{x:A} \mathsf{Id}_A\ a\ x)$

$\mathsf{ax}_5 : (\prod_{x:A} \mathsf{iscontr}\ (B(x))) \to \mathsf{iscontr}\ (\prod_{x:A} B(x))$

ax_6 : *The canonical map* $\mathsf{Id}_U\ X\ Y \to \mathsf{Weq}\ X\ Y$ *is a weak equivalence*

Model

Since the paper

D. Kan *A combinatorial definition of homotopy groups*, Annals of Mathematics, 1958, 67, 282-312

a convenient way to represent spaces is to use (Kan) simplicial sets

This forms a model of type theory (V. Voevodsky 2005, Th. Streicher 2006)

V. Voevodsky (2009) has extended this model to universes

This model satisfies (and suggested?) the univalence axiom

Computational interpretation

We have listed axiomatically some properties that the equality should have

All other notions in type theory are motivated/justified by computation rules

For instance

$$\mathsf{natrec} : P(0) \to (\prod_{x:N} P(x) \to P(x+1)) \to \prod_{x:N} P(x)$$

is justified by $\mathsf{natrec}\ a\ f\ 0 = a$ and $\mathsf{natrec}\ a\ f\ (n+1) = f\ n\ (\mathsf{natrec}\ a\ f\ n)$

(This represents at the same time both induction and recursion)

Can we justify in a similar way these axioms for equality?

Gandy's interpretation

On the Axiom of Extensionality

R. Gandy, The Journal of Symbolic Logic, 1956

Interpret $extensional$ type theory in $intensional$ type theory

The intuition is precisely that in λ-calculus a function can only occur in a proposition through its values in a term (cf. Russell's formulation of the axiom of extensionality)

This is only valid for $closed$ λ-terms: if X is a functional variable f does not appear in $X\ f$ through its values

Gandy's interpretation

The second part of the paper shows that a similar interpretation works for set theory

The paper is one of the first instance of the *logical relation* technique

We need to extend this technique to dependent types

Remark on Russell's work on implication

Russell *The Theory of Implications* 1906, American Journal of Mathematics

Russell does something similar but only for the axiom of extensionality on propositions

This amounts to show that all connectives preserve equivalences

Gandy's interpretation

One goal (current work) is to adapt Gandy's interpretation to dependent types

Intuitively: we know what the equality should be on all base types (on the universe U it should be weak equivalence) and so we can define equality on each type by induction on the types

This is similar to the work on *observational type theory* (Th. Altenkirch, C. McBride) and on *two-level type theory* (M. Maietti, G, Sambin) but generalizes them to the case of computationally relevant identity proofs

Invariance under isomorphisms: set theory/type theory

Lindenbaum and Tarski (1936): any provable formula is invariant under isomorphisms *in simple type theory*

Logical notions can be characterised as notions invariant under isomorphisms (Tarski, 1966)

This is *not valid* for set theory

Problem with *abstraction*: if we define a structure in set theory, properties of this structure are not close under isomorphims in general

Example: $X = \{0,1\},\ Y = \{a,b\}$ and the property is to contain 0

Invariance under isomorphisms: set theory/type theory

Sentences of Type Theory: The Only Sentences Preserved Under Isomorphisms
M.V. Marshall and R. Chaqui, *The Journal of Symbolic Logic*, 56, 1991, 932-948

Bourbaki calls structures defined by sentences preserved under isomorphisms "transportable"

In type theory, one expects that anything definable is invariant under isomorphisms (proved for simple type theory in the paper of Lindenbaum and Tarski 1936)

Some references

Th. Streicher, M. Hofmann *The groupoid interpretation of Type Theory* 1996

Home page of V. Voevodsky on *Univalent Foundation for Mathematics*

Work of S. Awodey and M. A. Warren

Some references

These works rely on special computation rules for equality

Nils Anders Danielsson has proved formally that we don't need new computation rules (axioms are enough)

This has been used crucially in this presentation

See www.cse.chalmers.se/~nad/listings/equality/README.html