Conversely, if we have a model $\langle \mathfrak{A}, R \rangle$ in $\mathscr{M}$, we let

$$W = \{\{a \mid \langle \mathfrak{A}, R \rangle \models Fa^*\} \mid \langle \mathfrak{A}, R \rangle \models ((\mathrm{Q}x\, Fx)^*\}.$$

It is straightforward that $\langle \mathfrak{A}, W \rangle$ is a countable weak LQ-model satisfying K1–K5 and LQ$\mathfrak{A}$ – elementary equivalent to $\langle \mathfrak{A}, R \rangle$. This gives

THEOREM 6. *For any* $F \in$ LQ, *the following are equivalent:*

(1) *F is true in all* LQ-*models.*
(2) *F is true in all weak* LQ-*models satisfying* K1–K5.
(3) *F is derivable from* K1–K5 *in first-order logic.*
(4) F^* *is true in all models in* $\mathscr{M}$.
(5) F^* *is derivable from* M1–M5.

This gives Result C.

References

[1] H. J. Keisler, Logic with the quantifier 'there exist uncountable many', *Ann. Math. Logic* 1 (1970) 1–93.
[2] G. Fuhrken, Languages with the added quantifier 'there exist at least $\aleph_\alpha$', in: *The theory of models*, J. W. Addison, L. Henkin and A. Tarski, Eds. (North-Holland, Amsterdam, 1967) pp. 121–131.

ABOUT MODELS FOR INTUITIONISTIC TYPE THEORIES AND THE NOTION OF DEFINITIONAL EQUALITY

Per MARTIN-LÖF

University of Stockholm, Stockholm, Sweden

0. Introduction

This paper consists of two parts. The first is devoted to the formulation of what seems to me to be the most natural notion of model for intuitionistic theories which either are type theories by their very definition, or else may be viewed as such because of the correspondence between formulae and type symbols discovered by Curry and Feys [2] and Howard [9]. The second analyzes the notion of definitional equality and its formal counterpart, convertibility, and advocates a change in the current definition of convertibility for systems in which explicit definitions are represented by means of lambda abstraction rather than the introduction of constants or the special constants called combinators. Because of the correspondence between lambda terms and natural deductions, this change is equally called for in Prawitz's definition of convertibility [19] (or equivalence, as he says) for natural deductions.

1. Models

The notion of model with which we shall be concerned can be formulated at least for

(1) the positive implicational calculus,
(2) intuitionistic propositional logic,

(3) intuitionistic first order predicate logic,
(4) the system of primitive recursive functions,
(5) primitive recursive arithmetic,
(6) intuitionistic first order arithmetic,
(7) the system of primitive recursive functionals of finite type,
(8) Gödel's theory T,
(9) intuitionistic arithmetic of finite type,
(10) intuitionistic ramified analysis,
(11) intuitionistic theories of generalized inductive definitions as formalized by Kreisel and Troelstra [14], Howard [11] and Martin-Löf [16],
(12) the intuitionistic theory of types of Martin-Löf [17],
(13) the system F of Girard [6] or, what amounts to essentially the same, intuitionistic second order logic with 0-ary predicate variables only,
(14) the theory of species,
(15) intuitionistic simple type theory.

The most important omissions in this list are systems containing axioms for choice sequences such as those of Kleene and Vesley [12] and Kreisel and Troelstra [14].

In the study of models of intuitionistic theories, one has the choice between classical and intuitionistic abstractions on the metalevel. Examples of classically described notions of model are the algebraic and topological interpretations, the Beth and Kripke semantics, Läuchli's abstract notion of realizability and the models of Stenlund [20] and Girard [7]. Examples of intuitionistic models are Kleene's realizability interpretation and the closely related model of convertible terms, first constructed by Tait [21] for Gödel's theory T. An obstacle to the formulation of a general intuitionistic notion of model has been the lack of a sufficiently welldeveloped intuitionistic notion of set.

Using the type-theoretic abstractions described in [17], I intend in the following to formulate an intuitionistic notion of model which is applicable to any one of the theories listed above and which is wide enough to include the realizability interpretation as well as the term model of the theory in question.

The transition to intuitionistic abstractions on the metalevel is both

essential and nontrivial. Essential, because in what seems to me to be the most fruitful notion of model, the interpretation of the convertibility relation conv, is standard, that is, it is interpreted as definitional equality $=_{\mathrm{def}}$ in the model, and definitional equality is a notion which is unmentionable within the classical set theoretic framework. Nontrivial, because of certain novelties which I would like to exemplify at once.

In the realizability interpretation, when described classically, one puts

$\bar{A} =_{\mathrm{def}}$ the set of natural numbers that realize the formula or type symbol A,

$$\mathrm{Ap}\,(e, m) \underset{\mathrm{def}}{=} \{e\}\,(m),$$

where, in the latter definition, it is supposed that m and e are natural numbers that realize A and $A \to B$, respectively. Intuitionistically, this no longer works, because there is no function in the intuitionistic sense which takes m and e into $\{e\}(m)$. Instead, we have to put

$\bar{A} \underset{\mathrm{def}}{=}$ the species of natural numbers that realize A,

$\mathrm{Obj}\,(\bar{A}) =_{\mathrm{def}} (\Sigma m \in \mathbf{N})\, \bar{A}(m) =_{\mathrm{def}}$ the type of pairs whose first component is a natural number m and whose second component is a proof that m realizes A,

$$\mathrm{Ap}\,(b, a) \underset{\mathrm{def}}{=} p\,(q\,(b, a)).$$

Here it is supposed that a and b are objects of types $\mathrm{Obj}\,(\bar{A})$ and $\mathrm{Obj}\,(\overline{A \to B})$, respectively, and that

$$e \text{ realizes } A \to B \underset{\mathrm{def}}{=} (\forall x \in \mathrm{Obj}\,(\bar{A}))\,(\exists y \in \mathrm{Obj}\,(\bar{B}))\,(\{e\}\,(p(x)) \simeq p(y))$$

which is logically equivalent (but not definitionally equal) to the more usual form

$$(\forall m \in N)\,(m \text{ realizes } A \to (\exists n \in \mathbf{N})\,(n \text{ realizes } B \;\&\; \{e\}(m) \simeq n)).$$

The functions p and q are the left and right projections of types

$$(\Sigma x \in A)\, B(x) \to A \quad \text{and} \quad (\Pi z \in (\Sigma x \in A)\, B(x))\, B\,(p(z)),$$

respectively, which are defined by the schema

$$\begin{cases} p\,((a, b)) =_{\mathrm{def}} a, \\ q\,((a, b)) =_{\mathrm{def}} b, \end{cases}$$

Σ being replaced by $\exists$ if we think of $B(a)$ for a of type A as a proposition rather than a type.

Similarly, in the term model, when described classically, one puts

$$\bar{A} \underset{\text{def}}{=} \text{the set of closed normal terms with type symbol } A,$$

Ap $(b, a) =_{\text{def}}$ the normal form of $b(a)$ which exists and is unique by virtue of the normalization theorem and the Church–Rosser property.

Again, this does not work intuitionistically, because the normal form of $b(a)$ is not a function of a and b alone. Instead, we have to put

$$\bar{A} \underset{\text{def}}{=} C_A \underset{\text{def}}{=} \text{the species of computable terms with type symbol } A,$$

Obj $(\bar{A}) =_{\text{def}} (\Sigma a \in \text{Term}\,(A))\, C_A(a) =_{\text{def}}$ the type of pairs whose first component is a closed term a with type symbol A and whose second component is a proof that a is computable,

$$\text{Ap}\,(b, a) \underset{\text{def}}{=} p\,(q\,(b, a)).$$

Here a and b are objects of types Obj $(\bar{A})$ and Obj $(\overline{A \to B})$, respectively, and

$$C_{A \to B}(b) \underset{\text{def}}{=} (\forall x \in \text{Obj}\,(\bar{A}))\,(\exists y \in \text{Obj}\,(\bar{B}))\,(b\,(p(x)) \text{ red } p(y))$$

which is logically equivalent to

$$(\forall a \in \text{Term}\,(A))\,(C_A(a) \to (\exists d \in \text{Term}\,(B))\,(C_B(d) \,\&\, b(a) \text{ red } d)).$$

Since the number of clauses needed in order to define what is a model of a certain theory grows in proportion to the number of clauses that specify the theory in question, I shall limit myself from now on to the positive implicational calculus and intuitionistic second order logic with 0-ary predicate variables only. Having given the complete definition for these, it should be clear how to extend it to the other theories listed above.

1.1. *The positive implicational calculus*

A *model* for the positive implicational calculus consists of the following data.

(a) A type Typ.

(b) A function Obj which to an arbitrary object A of type Typ assigns a type Obj (A).

(c) A function F of type Typ $\to$ Typ $\to$ Typ. Here and in the following parentheses are associated to the right. When there is no risk of confusion, I shall allow myself to write $A \to B$ instead of $F(A, B)$.

(d) A function Ap of type Obj $(F(A, B)) \to$ Obj $(A) \to$ Obj (B) for every pair of objects A and B of type Typ. Ap $(\cdots$ Ap $($Ap $(b, a_1), a_2) \ldots, a_n)$ will be abbreviated $b\,(a_1, \ldots, a_n)$.

(e) Closure under explicit definitions. For every finite sequence of objects $A_1, \ldots, A_n$ and B of type Typ and every term $b\,[x_1, \ldots, x_n]$ of type Obj (B) built up from variables $x_1, \ldots, x_n$ of types Obj $(A_1), \ldots$, Obj (A_n), respectively, by means of the operation Ap, there shall exist an object

$$f \in \text{Obj}\,(A_1 \to \cdots \to A_n \to B)$$

such that

$$f(a_1, \ldots, a_n) \underset{\text{def}}{=} b\,[a_1, \ldots, a_n].$$

By the combinatorial completeness property, it suffices in fact to have, for every triple of objects A, B and C of type Typ, objects I, K and S of types Obj $(A \to A)$, Obj $(A \to B \to A)$ and Obj $((A \to B \to C) \to (A \to B) \to A \to C)$, respectively, such that

$$I(a) \underset{\text{def}}{=} a,$$

$$K\,(a, b) \underset{\text{def}}{=} a,$$

$$S\,(c, b, a) \underset{\text{def}}{=} c\,(a, b(a)).$$

Besides the use of the intuitionistic type theoretic abstractions instead of the classical set theoretic ones on the metalevel, the most important difference between this notion of model and the models defined by Stenlund [20] and Girard [7] is the requirement that the equality in the equation $f(a_1, \ldots, a_n) =_{\text{def}} b\,[a_1, \ldots, a_n]$ be definitional and not merely set-theoretic equality or equality with respect to some arbitrary equivalence relation.

Suppose now that we are given an assignment of an object $\bar{A}$ of type Typ to every atomic formula A of the positive implicational calculus. Extend this assignment to composite formulae by putting $\overline{A \to B} =_{\text{def}} F(\bar{A}, \bar{B})$. In classical model theory one verifies that a formula which is formally derivable is true in an arbitrary model of the theory. For us, this step corresponds to showing how to assign to a closed term a with type symbol A an object $\bar{a}$ of type Obj $(\bar{A})$. The definition of $\bar{a}$ is by induction on the construction of a. However, during the induction we have to consider open terms as well. We put

$\bar{x} =_{\text{def}}$ a variable of type Obj $(\bar{A})$, provided x is a variable with type symbol A, $\qquad \overline{b(a)} \underset{\text{def}}{=} \text{Ap}\,(\bar{b}, \bar{a})$,

$\bar{f} =_{\text{def}}$ the object of type Obj $(\bar{A}_1 \to \cdots \to \bar{A}_n \to \bar{B})$ such that $\bar{f}(a_1, \ldots, a_n) =_{\text{def}} \bar{b}\,[a_1, \ldots, a_n]$ for $a_1, \ldots, a_n$ of types Obj $(\bar{A}_1), \ldots,$ Obj $(\bar{A}_n)$, respectively, provided f is the constant with type symbol $A_1 \to \cdots \to A_n \to B$ introduced by the schema $f(a_1, \ldots, a_n)$ conv $b\,[a_1, \ldots, a_n]$.

The assignment of $\bar{a}$ to a is clearly such that,

$$\text{if } a \text{ conv } b, \text{ then } \bar{a} \underset{\text{def}}{=} \bar{b}.$$

Thus the interpretation of the convertibility relation is standard.

1.1.1. Example. Intended interpretation.

(a) Typ $=_{\text{def}}$ the type of propositions.
(b) Obj $(A) =_{\text{def}}$ the type of proofs of the proposition A.
(c) $F(A, B) =_{\text{def}}$ the proposition A implies B.
(d) Ap $=_{\text{def}}$ modus ponens.
(e) Closure under explicit definitions is trivially fulfilled.

1.1.2. Example. Realizability interpretation.

(a) Typ $=_{\text{def}}$ the type of species of natural numbers.
(b) Obj $(A) =_{\text{def}} (\Sigma m \in \mathrm{N})\, A(m)$.

(c) $F(A, B) =_{\text{def}}$ the species of all natural numbers e such that

$$(\forall x \in \text{Obj}\,(A))\,(\exists y \in \text{Obj}\,(B))\,(\{e\}\,(p(x)) \simeq p(y)).$$

(d) Ap $(b, a) =_{\text{def}} p\,(q\,(b, a))$.
(e) Closure under explicit definitions is verified in very much the same way as in the term model which we shall construct next.

1.1.3. Example. Term model.

(a) Typ $=_{\text{def}}$ the type of pairs (A, φ), where A is a type symbol and φ a species of closed terms with type symbol A.
(b) Obj $((A, \varphi)) =_{\text{def}} (\Sigma a \in \text{Term}\,(A))\, \varphi(a)$.
(c) $F((A, \varphi), (B, \psi)) =_{\text{def}} (A \to B$, the species of all closed terms b with type symbol $A \to B$ such that

$$(\forall x \in \text{Obj}\,((A, \varphi)))\,(\exists y \in \text{Obj}\,((B, \psi)))\,(b\,(p(x)) \text{ red } p(y)).$$

(d) Ap $(b, a) =_{\text{def}} p\,(q\,(b, a))$.
(e) We shall verify closure under explicit definitions by considering a typical case, namely, we shall show how to interpret the constant K with type symbol $A \to B \to A$ which is defined by the schema

$$K(a, b) \text{ conv } a.$$

Its interpretation $\bar{K}$ is simply the pair consisting of the constant K and the usual proof that K is computable (see Tait [21]),

$$\bar{K} =_{\text{def}} (K, (\lambda x)\,((K\,(p(x)), (\lambda y)\,(x, \text{the proof that } K\,(p(x), p(y)) \text{ red } p(x))),$$
$$\text{the proof that } K\,(p(x)) \text{ red } K\,(p(x))))$$

where, of course, the use of the lambda notation is informal. For this $\bar{K}$ and a and b of types Obj $(\bar{A})$ and Obj $(\bar{B})$, respectively, we have

$$\bar{K}\,(a, b) \underset{\text{def}}{=} \text{Ap}\,(\text{Ap}\,(\bar{K}, a), b) \underset{\text{def}}{=} p\,(q\,(p\,(q\,(\bar{K}, a)), b)) \underset{\text{def}}{=} a$$

as desired. This finishes the construction of the term model for the positive implicational calculus.

1.2. *Intuitionistic second order logic*

We shall now extend the notion of model just introduced for the positive implicational calculus to the second order. A *model* for the fragment of intuitionistic second order logic in which only 0-ary predicate variables are allowed or, what amounts to essentially the same, the system F of Girard [6] consists of the following data.

(a) A type Typ.

(b) A function Obj which to an object A of type Typ assigns a type Obj (A).

(c) An assignment to every closed formula A of the extended language, obtained by adding the objects of type Typ as constants, of an object $\bar{A}$ of type Typ such that

$$\bar{A} \underset{\text{def}}{=} A$$

if A is (the constant for) an object of type Typ, and, furthermore, the *substitution property*

$$\bar{B}[\bar{A}] \underset{\text{def}}{=} \overline{B[A]}$$

is fulfilled. Observe that the equality here is definitional. In particular, the function F required for the positive implicational calculus is given by

$$F(A, B) \underset{\text{def}}{=} \overline{A \to B}$$

where, of course, $A \to B$ is the formula of the extended language obtained by applying the connective $\to$ to the constants A and B.

(d) For all closed formulae in the extended language of the forms $A \to B$ and $(\forall X)\, B[X]$, there shall be functions

$$\mathrm{Ap} \in \mathrm{Obj}\,\big(\overline{A \to B}\big) \to \mathrm{Obj}\,(\bar{A}) \to \mathrm{Obj}\,(\bar{B}),$$

$$\mathrm{Ap} \in \mathrm{Obj}\,\big(\overline{(\forall X)\, B[X]}\big) \to (\Pi X \in \mathrm{Typ})\, \mathrm{Obj}\,(\bar{B}[X]),$$

respectively.

(e) Closure under explicit definitions. If

$$(\forall X_1) \cdots (\forall X_m)\, (B_1 \to \cdots \to B_n \to C)$$

is a closed second order formula, and $c\,[X_1, \ldots, X_m, y_1, \ldots, y_n]$ is a term of type Obj $(\bar{C}\,[X_1, \ldots, X_m])$ built up from variables $X_1, \ldots, X_m$ of type Typ, terms of type Typ of the form $\bar{A}\,[X_1, \ldots, X_m]$, where $A\,[X_1, \ldots, X_m]$ is a second order formula, and variables $y_1, \ldots, y_n$ of types

$$\mathrm{Obj}\,(\bar{B}_1\,[X_1, \ldots, X_m]), \ldots, \mathrm{Obj}\,(\bar{B}_n\,[X_1, \ldots, X_m]),$$

respectively, by means of the functions Ap, there shall exist an object

$$f \in \mathrm{Obj}\,\big(\overline{(\forall X_1) \cdots (\forall X_m)\, (B_1 \to \cdots \to B_n \to C)}\big)$$

such that

$$f(A_1, \ldots, A_m, b_1, \ldots, b_n) \underset{\text{def}}{=} c\,[A_1, \ldots, A_m, b_1, \ldots, b_n].$$

Here it has been assumed, for notational simplicity, that all the universal quantifiers precede all the implications in the formula

$$(\forall X_1) \cdots (\forall X_m)\, (B_1 \to \cdots \to B_n \to C).$$

In general, they may occur in an arbitrary order.

The mapping of a closed term a with type symbol A into an object $\bar{a}$ of type Obj $(\bar{A})$ already defined for the positive implicational calculus is extended in the obvious way to the second order. That is, we add the new clause

$$\overline{b(A)} \underset{\text{def}}{=} \mathrm{Ap}\,(\bar{b}, \bar{A})$$

and change the third of the previous clauses to

$\bar{f} =_{\text{def}}$ the object of type Obj $\big(\overline{(\forall X_1) \cdots (\forall X_m)\, (B_1 \to \cdots \to B_n \to C)}\big)$ such that $\bar{f}(A_1, \ldots, A_m, b_1, \ldots, b_n) =_{\text{def}} \bar{c}\,[A_1, \ldots, A_m, b_1, \ldots, b_n]$ which we have required to exist provided f is a constant with type symbol $(\forall X_1) \cdots (\forall X_m)\, (B_1 \to \cdots \to B_n \to C)$ introduced by the schema $f(A_1, \ldots, A_m, b_1, \ldots, b_n)$ conv $c\,[A_1, \ldots, A_m, b_1, \ldots, b_n]$.

There is, however, one essential novelty that arises and which was overlooked by Stenlund [20]. Namely, we have to verify that $\overline{b(A)} = \mathrm{Ap}(\bar{b}, \bar{A})$ is an object of type Obj $(\overline{B[A]})$ provided $\bar{A}$ and $\bar{b}$ are the objects of types Typ and Obj $\big(\overline{(\forall X)\, B[X]}\big)$ associated with the formula A and the term b with type symbol $(\forall X)\, B[X]$, respectively. One sees immediately that $\overline{b(A)}$ is an object of type Obj $(\bar{B}[\bar{A}])$. But $\bar{B}[\bar{A}] =_{\text{def}} \overline{B[A]}$ by the substitution property and hence $\overline{b(A)}$ is indeed an object of type Obj $\big(\overline{B[A]}\big)$. Note that the last step in the argument is an application of the informal counterpart of the formal rule of type conversion formulated in [17].

Just as in the case of the positive implicational calculus, it is clear that the interpretation of the convertibility relation is standard, that is, that a conv b implies $\bar{a} =_{\text{def}} \bar{b}$.

1.2.1. EXAMPLE. Intended interpretation. Typ and Obj are defined as in the case of the positive implicational calculus. Moreover, for every formula A of the extended language, we put

$$\bar{A} \underset{\text{def}}{=} \text{the proposition denoted by } A.$$

We take the first of the functions Ap to be modus ponens just as before and the second to be

$$\text{Ap} \underset{\text{def}}{=} \text{universal instantiation.}$$

Closure under explicit definitions is trivially fulfilled.

1.2.2. EXAMPLE. Realizability interpretation. The extension to the second order is due to Kreisel and Troelstra [14].

(a) Typ $=_{\text{def}}$ the type of species of natural numbers.

(b) Obj $(A) =_{\text{def}} (\Sigma m \in \mathrm{N})\, A(m)$.

(c) For a closed formula A of the extended language, the definition of the species $\bar{A}$ is by induction on the construction of A.

If A is a constant, then $\bar{A}$ is the object of type Typ which it is a constant for.

$$\overline{A \to B}(e) \underset{\text{def}}{=} (\forall x \in \text{Obj}\,(\bar{A}))\,(\exists y \in \text{Obj}\,(\bar{B}))\,(\{e\}\,(p(x)) \simeq p(y)).$$

$$\overline{(\forall X)\, B[X]}(e) \underset{\text{def}}{=} (\forall X \in \text{Typ})\,(\exists y \in \text{Obj}\,(\bar{B}[X]))\,(e = p(y))$$

which is equivalent (but not definitionally equal) to the definition of Kreisel and Troelstra [14].

The verification of the substitution property $\bar{B}[\bar{A}] = \overline{B[A]}$ is immediate by induction on the construction of the formula $B[X]$.

(d) The first of the functions Ap is defined just as in the case of the positive implicational calculus and the second by

$$\text{Ap}\,(b, A) \underset{\text{def}}{=} p\,(q\,(b, A)).$$

(e) Closure under explicit definitions. Let the constant f of type

$$(\forall X_1) \cdots (\forall X_m)\,(B_1 \to \cdots \to B_n \to C)$$

be introduced by the schema

$$f(A_1, \ldots, A_m, b_1, \ldots, b_n) \text{ conv } c\,[A_1, \ldots, A_m, b_1, \ldots, b_n],$$

where $c\,[X_1, \ldots, X_m, y_1, \ldots, \ldots, y_n]$ is a derivation

$$\begin{array}{ccc} B_1\,[X_1, \ldots, X_m] & \cdots & B_n\,[X_1, \ldots, X_m] \\ \ddots & & \therefore \\ & C\,[X_1, \ldots, X_m] & \end{array}$$

with free variables and assumptions as indicated. In the proof that every derivable formula is realizable, one shows how to associate with such a derivation a Gödel number e and a proof that, for all species of natural numbers $A_1, \ldots, A_m$, if e_j realizes $B_j\,[X_1, \ldots, X_m]$ relative to $A_1, \ldots, A_m$ for $j = 1, \ldots, n$, then $\{e\}\,(e_1, \ldots, e_n)$ is defined and realizes $C[X_1, \ldots, X_m]$ relative to $A_1, \ldots, A_m$. The number e together with this proof is essentially the object $\bar{f}$ of type Obj $\big(\overline{(\forall X_1) \cdots (\forall X_m)\,(B_1 \to \cdots \to B_n \to C)}\big)$ that interprets the constant f. The verification that

$$\bar{f}(A_1, \ldots, A_m, b_1, \ldots, b_n) \underset{\text{def}}{=} \bar{c}\,[A_1, \ldots, A_m, b_1, \ldots, b_n]$$

will be omitted since it is completely analogous to the corresponding verification for the term model.

1.2.3. EXAMPLE. Term model. The extension to the second order is due to Girard [6].

(a) Typ $=_{\text{def}}$ the type of pairs of the form (A, φ), where A is a closed second order formula and φ a species of closed terms with type symbol A.

(b) Obj $((A, \varphi)) =_{\text{def}} (\Sigma a \in \text{Term}\,(A))\, \varphi(a)$, where Term (A) denotes the type of closed terms with type symbol A.

(c) For a closed formula A of the extended language, the definition of the object $\bar{A}$ of type Typ is by induction on the construction of A.

$$\bar{A} \underset{\text{def}}{=} A \text{ if } A \text{ is (the constant for) an object of type Typ.}$$

For an implication the definition is as in the case of the positive implicational calculus.

$\overline{(\forall X)\, B[X]} =_{\text{def}} ((\forall X)\, B[X]$, the species of all closed terms b with type symbol $(\forall X)\, B[X]$ such that

$$(\forall X \in \text{Typ})\, (\exists y \in \text{Obj}\, (\bar{B}[X]))\, (b(p(X)) \text{ red } p(y))).$$

Here parameters have been suppressed for the sake of notational simplicity.

The fact that $\bar{B}[\bar{A}] =_{\text{def}} \overline{B[A]}$, which is seen by induction on the construction of the formula $B[X]$, is essentially the content of the substitution lemma in Girard [6]. However, it has to be observed as in [15] that the equality in the substitution lemma is definitional and not merely extensional.

(d) Application.

$$\text{Ap}\,(b, a) \underset{\text{def}}{=} p\,(q\,(b, a)).$$

$$\text{Ap}\,(b, A) \underset{\text{def}}{=} p\,(q\,(b, A)).$$

(e) Closure under explicit definitions. Just as in the case of the positive implicational calculus, we shall verify this by considering a typical case. Let the constant I with type symbol $(\forall X)\,(X \to X)$ be defined by the schema

$$I\,(A, b) \text{ conv } b.$$

We construct its interpretation $\bar{I}$ of type $\text{Obj}\,\big(\overline{(\forall X)\,(X \to X)}\big)$ by taking the pair consisting of the constant I and the proof that I is computable,

$$\bar{I} =_{\text{def}} (I, (\lambda X)\, ((I\,(p(X)), (\lambda y)\, (y, \text{the proof that } I\,(p(X), p(y)) \text{ red } p(y))), \text{the proof that } I\,(p(X)) \text{ red } I\,(p(X)))).$$

Then

$$\bar{I}\,(A, b) \underset{\text{def}}{=} \text{Ap}\,(\text{Ap}\,(\bar{I}, A), b) \underset{\text{def}}{=} p\,(q\,(p\,(q\,(\bar{I}, A)), b)) \underset{\text{def}}{=} b$$

for A of type Typ and b of type $\text{Obj}\,(A)$ as desired.

2. Definitional equality

By definitional equality, I mean the relation which is used on almost every page of an informal mathematical text and which is denoted by $\equiv$, $=_{\text{def}}$ or most often but less felicitously simply $=$. As an example, one can take the first part of this paper where it has been used more than fifty times. Being informal, it occurs in the left column of the following dictionary which shows the relation between certain informal notions and their formal counterparts.

informal	formal
proposition	formula
proof	derivation, proof figure
type	type symbol
mathematical object	term
defining equation	rule of conversion
definiendum	redex
definiens	contractum
definitional equality	convertibility

Thus the formal counterpart of definitional equality is the relation of convertibility studied in combinatory logic and proof theory.

Definitional equality is a relation between linguistic expressions and *not* between the abstract entities which they denote and which are the same. This is the view that Frege [3] took of the relation of equality of content (Inhaltsgleichheit) which enters into his Begriffsschrift but which he later abandoned.

I claim that the relation of definitional equality is determined by the following three principles and by these principles alone.

(i) A definiens is always definitionally equal to its definiendum.
(ii) Definitional equality is preserved under substitution. That is, if we substitute two definitionally equal expressions for a variable in a given expression, then the resulting expressions are also definitionally equal.
(iii) Definitional equality is an equivalence relation, that is, it is reflexive, symmetric and transitive.

This claim is supported by the following heuristic evidence. The only place where the relation of definitional equality is used in a crucial way except in the definitional schemata themselves is in arguments of the form

if a is an object of type A and $A =_{\text{def}} B$, then a is an object of type B, and, correspondingly for propositions and proofs,

if a is a proof of the proposition A and $A =_{\text{def}} B$, then a is a proof of the proposition B.

This principle is accepted on the basis that if $A =_{\text{def}} B$, then A and B are merely notational variants of one and the same abstract type or proposition, as the case may be. Detailed case studies show that the relation of definitional equality with respect to which this principle is applied has to satisfy precisely the above three conditions. Here is a typical example.

Define a type valued function F by the schema

$$\begin{cases} F(0) =_{\text{def}} \mathbf{N}, \\ F(n+1) =_{\text{def}} F(n) \to F(n). \end{cases}$$

Then, given a function f of type $(\Pi n \in \mathbf{N})\, F(n)$, we can define a function g of the same type by putting

$$g(n) \underset{\text{def}}{=} f(n+1, f(n)).$$

Indeed, if n is an arbitrary natural number, $f(n)$ and $f(n+1)$ are objects of types $F(n)$ and $F(n+1)$, respectively. But $F(n+1) =_{\text{def}} F(n) \to F(n)$ *and hence* $f(n+1)$ *is a function of type* $F(n) \to F(n)$, so that we can apply it to $f(n)$, thereby getting an object $f(n+1, f(n))$ of type $F(n)$.

Let us now see what corresponds to the three principles determining the relation of definitional equality on the formal level. Clearly, they are turned into the *conversion rules*

$$\text{redex conv contractum}, \qquad \frac{a \text{ conv } c}{b[a] \text{ conv } b[c]},$$

$$a \text{ conv } a, \qquad \frac{a \text{ conv } b}{b \text{ conv } a}, \qquad \frac{a \text{ conv } b \quad b \text{ conv } c}{a \text{ conv } c},$$

in the second of which the terms a and c must have the same type symbol as the variable x in $b[x]$ for which they are substituted. The corresponding *reduction* relation is obtained by omitting the symmetry rule and the *strict reduction* relation by omitting the reflexivity as well.

In particular, in the positive implicational calculus (or, what amounts to the same, the basic theory of functionality in [2]) when formulated with constants, the conversion rules are equivalent to the following

$$f(a_1, \ldots, a_n) \text{ conv } b\,[a_1, \ldots, a_n],$$

$$\frac{a \text{ conv } c}{b(a) \text{ conv } b(c)}, \qquad \frac{b \text{ conv } d}{b(a) \text{ conv } d(a)},$$

$$a \text{ conv } a, \qquad \frac{a \text{ conv } b}{b \text{ conv } a}, \qquad \frac{a \text{ conv } b \quad b \text{ conv } c}{a \text{ conv } c}$$

which generate the convertibility and, if symmetry is left out, reduction relations which are called *weak* in combinatory logic. When only the special constants called combinators are allowed, the first rule of conversion specializes to

$$I(a) \text{ conv } a,$$

$$K(a, b) \text{ conv } a,$$

$$S(c, b, a) \text{ conv } c(a, b(a)).$$

On the other hand, if we consider the typed lambda calculus, which is isomorphic to the natural deduction formulation of the positive implicational calculus, then the above rules of conversion reduce to

$$(\lambda x)\, b[x]\,(a) \text{ conv } b[a], \qquad \frac{a \text{ conv } c}{b[a] \text{ conv } b[c]},$$

$$a \text{ conv } a, \qquad \frac{a \text{ conv } b}{b \text{ conv } a}, \qquad \frac{a \text{ conv } b \quad b \text{ conv } c}{a \text{ conv } c}.$$

The corresponding reduction relation, which is obtained by leaving out the symmetry rule, is precisely the *restricted* (as opposed to *general*) reduction relation introduced by Howard [10]. But, and this is the important point, the convertibility relation generated by these rules is *not* (the typed version of) the usual convertibility relation between lambda terms as defined in [1] and [2], because the rule

$$\frac{b[x] \text{ conv } d[x]}{(\lambda x)\, b[x] \text{ conv } (\lambda x)\, d[x]} \qquad (\xi)$$

which cannot be derived from the others, has been left out. Similarly, the reduction relation between natural deductions introduced by Prawitz [18] corresponds *not* to the restricted but to the general reduction relation, because in a reduction step as defined by him

$$\dfrac{\begin{matrix}\vdots\\ A\end{matrix} \quad \dfrac{\begin{matrix}A\\ \vdots\\ B\end{matrix}}{A \to B}}{\begin{matrix}B\\ \vdots\end{matrix}} \quad \text{red} \quad \begin{matrix}\vdots\\ A\\ \vdots\\ B\\ \vdots\end{matrix}$$

there may be open assumptions in the subderivation

$$\dfrac{\begin{matrix}\vdots\\ A\end{matrix} \quad \dfrac{\begin{matrix}A\\ \vdots\\ B\end{matrix}}{A \to B}}{B}$$

which become closed (discharged or cancelled) further down in the derivation, that is, below the downmost occurrence of the formula *B*.

The outcome of the foregoing analysis is that the rule (ξ) is unacceptable as a rule of conversion. Of course, we are free to define many different relations between terms and call them convertibility relations, but my claim is that only one of these correctly formalizes the informal notion of definitional equality. And a correct definition of convertibility is vital in all systems, in particular, in all higher type systems like Gödel's *T*, intuitionistic arithmetic of finite type and intuitionistic

simple type theory, whose formulae alias type symbols are not necessarily in normal form, because we then need the rule of inference

$$\frac{A}{B} \quad A \text{ conv } B$$

alias the rule of term formation

if *a* is a term with type symbol *A* and *A* conv *B*, then *a* is a term with type symbol *B*.

Hence a change in the definition of convertibility may change the stock of derivations alias terms of the theory and even the derivability relation. This difficulty does not arise in systems whose formulae alias type symbols are all in normal form, because then the derivations alias terms can be generated separately, that is, without reference to the convertibility relation, whose definition can wait until afterwards. Examples of such systems are intuitionistic first order predicate logic and Girard's system *F*.

2.1. *Positive effects of abolishing the conversion rule* (ξ)

2.1.1. For the models described in the first part of this paper, we achieve that *a* conv *b* implies $\bar{a} =_{\text{def}} \bar{b}$. In particular, in the realizability interpretation, we achieve that, if *a* and *b* are two interconvertible derivations of the formula *A*, then the corresponding numbers which realize *A* as well as the proofs which show that they do so are definitionally equal. Similarly, in the term model, we achieve that, if *a* conv *b*, then the normal forms of *a* and *b* as well as the proofs which show that they are computable (hereditarily normalizable) are definitionally equal.

2.1.2. When defining his notion of model for functional systems up to the level of intuitionistic simple type theory, Girard [7] introduces a relation reduction $*$ between closed terms of an extended language obtained by adding as new constants elements of certain sets of arbitrary cardinality. The relation reduction $*$ is obtained from the usual reduction relation for lambda terms by not allowing a redex to be contracted unless it is closed. Now, since we are only dealing with closed terms, this is precisely the same restriction as the one that has been advocated above.

Girard [7] also notes that the realizability interpretation alias the model of the hereditarily recursive operations is *not* a model with respect to the general reduction relation for lambda terms. This is his reason for introducing the restricted relation reduction $*$.

2.1.3. For the natural transformations from lambda terms to terms built up from constants or combinators and vice versa, denoted by the superscripts $^{\circ}$ and $^{\bullet}$, respectively, we achieve that, for lambda terms a and b

$$a \text{ conv } b \text{ implies } a^{\circ} \text{ conv } b^{\circ}$$

while preserving the property that, for terms a and b built up from constants or combinators,

$$a \text{ conv } b \text{ implies } a^{\bullet} \text{ conv } b^{\bullet}.$$

The transformations $^{\circ}$ and $^{\bullet}$ are defined as follows.

2.1.3.1. For a variable in the lambda calculus, we put

$$x^{\circ} \underset{\text{def}}{=} x.$$

Furthermore,

$$(b(a))^{\circ} \underset{\text{def}}{=} b^{\circ}(a^{\circ}),$$

$$((\lambda x)\, b\,[a_1, \ldots, a_n, x])^{\circ} \underset{\text{def}}{=} f(a_1^{\circ}, \ldots, a_n^{\circ}),$$

where $a_1, \ldots, a_n$ are the (necessarily disjoint) maximal subterms of $b\,[a_1, \ldots, a_n, x]$ that do not contain any free occurrences of the variable x, and f is the function constant introduced by the schema

$$f(a_1, \ldots, a_n, a) \text{ conv } b^{\circ}\,[a_1, \ldots, a_n, a].$$

If we only allow the special constants I, K and S, we have to put instead

$$((\lambda x)\, b[x])^{\circ} \underset{\text{def}}{=} (\lambda x)\, b^{\circ}[x],$$

where

$$(\lambda x)\, x \underset{\text{def}}{=} I,$$

$$(\lambda x)\, a \underset{\text{def}}{=} K(a)$$

provided x does not occur free in a, and

$$(\lambda x)\,(b[x]\,(a[x])) \underset{\text{def}}{=} S\,((\lambda x)\, b[x], (\lambda x)\, a[x])$$

provided x occurs free in at least one of the terms $a[x]$ and $b[x]$ (otherwise the previous clause is applicable).

2.1.3.2. Conversely,

$$x^{\bullet} \underset{\text{def}}{=} x,$$

$$(b(a))^{\bullet} \underset{\text{def}}{=} b^{\bullet}(a^{\bullet})$$

and, if f is a function constant introduced by the schema

$$f(a_1, \ldots, a_n) \text{ conv } b\,[a_1, \ldots, a_n],$$

then

$$f^{\bullet} \underset{\text{def}}{=} (\lambda x_1) \cdots (\lambda x_n)\, b^{\bullet}\,[x_1, \ldots, x_n]$$

so that, in particular,

$$I^{\bullet} \underset{\text{def}}{=} (\lambda x)\, x,$$

$$K^{\bullet} \underset{\text{def}}{=} (\lambda x)\,(\lambda y)\, x,$$

$$S^{\bullet} \underset{\text{def}}{=} (\lambda z)\,(\lambda y)\,(\lambda x)\,(z\,(x, y(x))).$$

2.1.4. The proof of normalization for my intuitionistic type theory (see [17]) becomes locally formalizable in the theory itself. When the dubious rule of lambda conversion was allowed, I could not carry out the proof of normalization for every specific term in the theory itself, contrary to what one would expect from one's experience with other full scale formal theories. The reason for this failure was that, when A conv B in the old sense, I was only able to prove C_A and C_B to be extensionally equal, whereas one would like to have $C_A =_{\text{def}} C_B$. Here C_A and C_B are the computability predicates associated with the type symbols A and B, respectively.

2.1.5. By forbidding the rule

$$\frac{b[x] \text{ conv } d[x]}{(\lambda x)\, b[x] \text{ conv } (\lambda x)\, d[x]},$$

Howard [10] was able to achieve, for his unique assignment of ordinals to the terms of Gödel's T, that

$$\text{if } a \text{ reduces strictly to } b, \text{ then } \alpha > \beta,$$

where α and β are the ordinals $< \varepsilon_0$ assigned to the terms a and b, respectively. For general reductions, this property is no longer known to hold.

2.2. *Further rules of conversion which do not correctly formalize the notion of definitional equality as understood in this paper*

2.2.1. Curry's rule of η-conversion,

$$(\lambda x)\,(b(x)) \text{ conv } b$$

provided the variable x does not occur free in the term b, and the combinatory axioms which correspond to it. Equally unacceptable is the corresponding rule for cartesian products,

$$(p(c),\, q(c)) \text{ conv } c,$$

although, as shown below, the abstract objects denoted by $(p(c), q(c))$ and c can be proved to be *identical.*

2.2.2. In systems of natural deduction, the following rules which are all formulated in [19]. First, the *permutative* rules for $\vee$ and $\exists$,

$$\dfrac{\dfrac{\begin{matrix}\vdots\\ A\vee B\end{matrix}\quad \begin{matrix}A\\ \vdots\\ C\end{matrix}\quad \begin{matrix}B\\ \vdots\\ C\end{matrix}}{C}\quad \vdots}{D}$$

$$\text{conv}\quad \dfrac{\begin{matrix}\vdots\\ A\vee B\end{matrix}\quad \dfrac{\begin{matrix}A\\ \vdots\\ C\end{matrix}\quad \vdots}{D}\quad \dfrac{\begin{matrix}B\\ \vdots\\ C\end{matrix}\quad \vdots}{D}}{D}$$

and

$$\dfrac{\dfrac{\begin{matrix}\vdots\\ (\exists x)\, B[x]\end{matrix}\quad \begin{matrix}B[x]\\ \vdots\\ C\end{matrix}}{C}\quad \vdots}{D}\quad \text{conv}\quad \dfrac{\begin{matrix}\vdots\\ (\exists x)\, B[x]\end{matrix}\quad \dfrac{\begin{matrix}B[x]\\ \vdots\\ C\end{matrix}\quad \vdots}{D}}{D}$$

provided the inference from C to D neither binds any free variable nor discharges any assumption in the derivation of $A \vee B$ and $(\exists x)\, B[x]$, respectively. Second, the *simplification* rules which are used to get rid of redundant applications of the elimination rules for $\vee$ and $\exists$. Third, the *expansion* rules, one for each of the logical operations, which in the case of implication reads

$$\dfrac{\dfrac{\begin{matrix}\vdots\\ A\to B\end{matrix}\quad A}{B}}{A\to B}\quad \text{conv}\quad \begin{matrix}\vdots\\ A\to B\end{matrix}$$

and corresponds to Curry's rule of η-conversion.

2.3. *Definitional equality versus identity*

It is necessary to distinguish carefully between, on the one hand, the relation of definitional equality which, according to what has been said above, is a relation between linguistic expressions and, on the other hand, the relation of identity between the abstract entities that they denote.

I regard the identity relation $a = b$ between objects a and b of some type A as defined by the axiom of $=$-introduction

$$a = a$$

alias the object

$$r(a) \text{ of type } a = a$$

analogously to the way in which the logical operations &, $\vee$ and $\exists$ and the type **N** are defined by their respective introduction rules. The corresponding axiom of =-elimination is

$$(\forall x \in A)\, C\,[x, x, r(x)] \to (\forall z \in a = b)\, C\,[a, b, z]$$

which, in case the predicate $C\,[a, b, c]$ does not depend on the proof c of the proposition $a = b$, reduces to

$$(\forall x \in A)\, C\,[x, x] \to (a = b \to C\,[a, b)]\,.$$

This, in turn, is equivalent (modulo the axioms of implication and universal quantification) to the usual eliminatory axiom of identity

$$a = b \to (C[a] \to C[b]).$$

In one direction, the relation between definitional equality and identity is as follows.

If $a =_{\mathrm{def}} b$, then $a = b$ holds,

and, on the formal level,

if a conv b, then $a = b$ is derivable.

Informally, we argue that $a = a$ is an axiom and that $a =_{\mathrm{def}} b$ implies $(a = a) =_{\mathrm{def}} (a = b)$ so that $a = a$ and $a = b$ have the same meaning and we can conclude $a = b$. The last step in the argument amounts on the formal level to an application of the (indispensable) rule of formula alias type conversion formulated in [17].

In the other direction, there seems to be little hope of showing that, if $a = b$ holds, then $a =_{\mathrm{def}} b$, or even that, if a and b are terms of a formally delimited theory, the validity of $a = b$ should imply a conv b. Little hope, because to say that $a = b$ holds intuitionistically means only that we suppose that we have a completely arbitrary abstract proof of $a = b$, and it seems too much to hope for that we should be able to pass from such an abstract proof to the sequence of combinatorial transformations that would establish a conv b. However, if we assume not only that $a = b$ holds but that $a = b$ is derivable in a formally delimited theory, then we have the following precise answer.

THEOREM. *If there exists a closed derivation of $a = b$, then the terms a and b are interconvertible.*

PROOF. This follows, for any one of the theories listed in the first part of this paper, as a combinatorial corollary to the normalization theorem for the theory in question. Suppose namely that there exists a closed derivation of $a = b$. (Then the terms a and b are necessarily closed too.) Using the normalization theorem, we can reduce it to normal form. Now, a closed normal derivation must necessarily have introduction form. In particular, when the end formula is $a = b$, it must have the form

$$\frac{c = c}{a = b} \quad (a = b) \text{ conv } (c = c).$$

The assumption that there is precisely one application of the rule of formula conversion between the axiom $c = c$ and the end formula $a = b$ implies no essential restriction of generality, because, if there were several, we could condense them into one, and, if there were none, we could insert a redundant application of the rule in question. From $(a = b)$ conv $(c = c)$ the Church–Rosser theorem allows us to conclude that $a = b$ and $c = c$ have a common reduct. Hence so do, on the one hand, a and c, and, on the other hand, b and c. Therefore, the terms a and b are interconvertible as was to be proved.

The theorem is not so interesting as it may seem, in particular, it proves nothing about the adequacy of the definition of convertibility (cf. [13]), because the relation with respect to which we conclude that a and b are interconvertible is just the convertibility relation which we put into the theory via the rule of formula alias type conversion.

The following counterexample shows that the theorem is no longer true for open terms. Consider a cartesian product $A \times B$ and let p and q be the associated projections with type symbols $A \times B \to A$ and $A \times B \to B$, respectively, defined by the schema

$$\begin{cases} p\,((a, b)) \text{ conv } a, \\ q\,((a, b)) \text{ conv } b. \end{cases}$$

Then, for free variables x and y with type symbols A and B, respectively,

$$(x, y) = (x, y)$$

is an instance of the law of identity from which we can infer

$$(p\,((x, y)), q\,((x, y))) = (x, y)$$

by formula conversion. The latter formula taken together with the axiom

$$(\forall x \in A)\, (\forall y \in B)\, C\,[(x, y)] \to (\forall z \in A \times B)\, C[z]$$

yields

$$(p(z), q(z)) = z$$

for a free variable z with type symbol $A \times B$, although the term $(p(z), q(z))$ does not convert into z with respect to the above rules of conversion. On the other hand, if c is a *closed* term with type symbol $A \times B$, then $(p(c), q(c))$ conv c, because a closed term with type symbol $A \times B$ necessarily reduces to one of the form (a, b).

2.4. *Discussion of the conjecture about identity of proofs formulated by Prawitz* [19].

The conjecture was that two derivations represent the same proof if and only if they are equivalent. Here equivalent means interconvertible.

Clearly, the conjecture hinges upon what we understand by two proofs being the *same*. Two, and only two, interpretations seem possible.

Either we mean by saying that two proofs are the same that they are *definitionally equal* which, according to what has been said above, is an assertion about the proofs thought of as linguistic expressions. In that case, the conjecture is turned into the thesis which has been advocated above, namely, that the relation of convertibility as defined in this paper correctly formalizes the notion of definitional equality.

Or else we really have the abstract proofs in mind and not their linguistic representations. Then sameness must mean identity, and the conjecture is turned into the assertion that two derivations are interconvertible if and only if the abstract proofs that they represent are identical. As was argued above, there seems to be little hope of proving the conjecture in this form unless identical is replaced by provably identical in which case the theorem and the remarks following it give a complete answer.

2.5. *On the treatment of equality in Frege's writings*

As was mentioned earlier, equality appears in §8 of Frege's Begriffsschrift as equality of content (Inhaltsgleichheit) which he denotes by $\equiv$ and which is a relation between names and not between their contents. It seems reasonable to identify Frege's equality of content (provided one disregards the geometrical example that he gives) with definitional equality

or, on the formal level, convertibility as understood in the present paper.

So far so good, but later, in §20 and §21, the axioms of identity are written (in modern notation)

$$a \equiv b \rightarrow (A(a) \rightarrow A(b)),$$
$$a \equiv a.$$

This is no longer compatible with the analysis of the relation $\equiv$ given earlier, because if $\equiv$ is viewed as a relation between names, then $a \equiv b$ is not a proposition on a par with the propositions inside the formal theory like $A(a)$ and $A(b)$ which we prove by means of possibly logically complicated proofs. In particular, it cannot be combined with these into compound propositions by means of the logical operations. Thus, for example, $a \equiv b \rightarrow (A(a) \rightarrow A(b))$ is meaningless because in $a \equiv b$ the entities a and b are names, that is, they stand for themselves, whereas in $A(a)$ and $A(b)$ they stand for their contents. This caused Frege [4, 5] to abandon the relation of equality of content $\equiv$ and replace it by the relation of identity $=$.

Similarly, something like

$$\text{for all natural numbers } n,\ n \underset{\text{def}}{=} n$$

is meaningless and, accordingly, $(\forall x \in \mathbf{N})$ (x conv x) is not a wellformed formula, because the variable x ranges over the natural numbers and not over the numerical terms of some formal theory. On the other hand,

$$\text{for all natural numbers } n,\ n = n$$

is a meaningful and true proposition which is expressed by the formula $(\forall x \in \mathbf{N})\,(x = x)$. Also meaningful and true is the proposition

$$\text{for all numerical terms } a,\ a \text{ conv } a,$$

but it can only be expressed in the formal theory after arithmetization.

2.6. *Equality in Gödel's T*

The relation of equality enters into Gödel's theory T in two different ways, on the one hand in the definitional schemata of the primitive recursive functionals, and on the other in the associated deductive theory. In the definitional schemata, the equality relation is clearly definitional which implies that they should be written formally

$$f(a_1, \dots, a_n) \text{ conv } b\,[a_1, \dots, a_n],$$

$$\begin{cases} f(a_1, \dots, a_n, 0) \text{ conv } a\,[a_1, \dots, a_n], \\ f(a_1, \dots, a_n, a') \text{ conv } b\,[a_1, \dots, a_n, a, f(a_1, \dots, a_n, a)]. \end{cases}$$

In the deductive theory, on the other hand, the equality relation has to be understood as identity and not as intensional or definitional equality as suggested by Gödel [8], because there we prove equalities by means of the axioms of identity

$$a = a, \quad \frac{a = b \quad A[a]}{A[b]},$$

and the induction schema

$$\frac{A[0] \quad \begin{matrix} A[x] \\ A[x'] \end{matrix}}{A[a]}$$

of whose validity we cannot convince ourselves unless, when reading the formulae, we associate with the terms not themselves but the abstract objects which they denote. To complete the formulation of the deductive part of the theory, we only have to add the rule of formula conversion

$$\frac{A}{B} \quad A \text{ conv } B.$$

If we allow as formulae not only equalities between terms of arbitrary finite type but also propositional combinations of such, then we shall have to add the rules of *intuitionistic* propositional logic, because we have no right to assert

$$(a = b) \vee -(a = b)$$

intuitionistically except at the lowest type in which case it is derivable from the other axioms, provided $-A$ is defined as $A \to 0 = 1$.

This should be compared with the fact that, as proved by Tait [21] for all terms a and b of an arbitrary finite type,

$$(a \text{ conv } b) \vee -(a \text{ conv } b).$$

However, the decidability of the convertibility relation provides no evidence whatever for the decidability of the identity relation, the former being a relation between terms and the latter a relation between the abstract objects which the terms denote.

The identity relation on an arbitrary type is decidable if and only if there exists a numerical valued equality functional E such that

$$E(a, b) = 0 \leftrightarrow a = b,$$

and hence we have just as little right to postulate the existence of such a functional as the decidability of the identity relation except at the lowest type where we can put $E(a, b) =_{\text{def}} |a - b|$. On the other hand, Tait's proof of the decidability of the convertibility relation provides us for every finite type with a function E such that

$$E(a, b) = 0 \leftrightarrow a \text{ conv } b.$$

However, this function E is defined for the *terms* and not for the abstract objects of the type in question, and hence it is not on a par with the functions that are defined by the ordinary definitional schemata of explicit definition and recursion.

Because of what has been said above, the system of intuitionistic arithmetic of finite type formulated by Tait [21] is not intuitionistically acceptable unless the axiom $(a = b) \vee -(a = b)$ is abolished at higher types. And, in the intensional version of the system formulated by Troelstra [22], not only this axiom but also the equality functional has to be thrown out.

Acknowledgments

I am grateful to Peter Hancock and Sören Stenlund for pointing out errors in the first draft of this paper.

References

[1] A. Church, *The calculi of lambda-conversion*, Annals of Mathematics Studies No. 6 (Princeton University Press, Princeton, N. J., 1941).
[2] H. B. Curry and R. Feys, *Combinatory logic*, Vol. I (North-Holland, Amsterdam, 1958).
[3] G. Frege, *Begriffsschrift, eine der arithmetischen nachgebildete Formelsprache des reinen Denkens* (Verlag von Louis Nebert, Halle, 1879).
[4] G. Frege, Über Sinn und Bedeutung, *Z. Philosophie und philosophische Kritik* 100 (1892) 25–50.
[5] G. Frege, *Grundgesetze der Arithmetik, begriffsschriftlich abgeleitet*, I. Band (Verlag von H. Pohle, Jena, 1893).
[6] J. Y. Girard, Une extension de l'interprétation de Gödel à l'analyse, et son application à l'élimination des coupures dans l'analyse et dans la théorie des types, in: *Proceedings of the second Scandinavian logic symposium*, J. E. Fenstad, Ed. (North-Holland, Amsterdam, 1971) pp. 63–92.
[7] J. Y. Girard, Interprétation fonctionnelle et élimination des coupures de l'arithmétique d'ordre supérieur, Thèse, Université Paris VII (1972).
[8] K. Gödel, Über eine bisher noch nicht benützte Erweiterung des finiten Standpunktes, *Dialectica* 12 (1958) 280–287.
[9] W. A. Howard, The formulae-as-types notion of construction, privately circulated notes.
[10] W. A. Howard, Assignment of ordinals to terms for primitive recursive functionals of finite type, in: *Intuitionism and proof theory*, J. Myhill, A. Kino and R. E. Vesley, Eds. (North-Holland, Amsterdam, 1970) pp. 443–458.
[11] W. A. Howard, A system of abstract constructive ordinals, *J. Symb. Logic* 37 (1972) 355–374.
[12] S. C. Kleene and R. E. Vesley, *The foundations of intuitionistic mathematics* (North-Holland, Amsterdam, 1965).
[13] G. Kreisel, A survey of proof theory II, in: *Proceedings of the second Scandinavian logic symposium*, J. E. Fenstad, Ed. (North-Holland, Amsterdam, 1971) pp. 109–170.
[14] G. Kreisel and A. S. Troelstra, Formal systems for some branches of intuitionistic analysis, *Ann. Math. Logic* 1 (1970) 229–387.
[15] P. Martin-Löf, Hauptsatz for intuitionistic simple type theory, in: *Logic, methodology and philosophy of science IV*, P. Suppes et al. Eds. (North-Holland, Amsterdam, 1973) pp. 279–290.
[16] P. Martin-Löf, Hauptsatz for the intuitionistic theory of iterated inductive definitions, in: *Proceedings of the second Scandinavian logic symposium*, J. E. Fenstad, Ed. (North-Holland, Amsterdam, 1971) pp. 179–216.
[17] P. Martin-Löf, An intuitionistic theory of types, mimeographed.
[18] D. Prawitz, *Natural deduction, a proof-theoretical study* (Almqvist and Wiksell, Stockholm, 1965).

[19] D. Prawitz, Ideas and results in proof theory, in: *Proceedings of the second Scandinavian logic symposium*, J. E. Fenstad, Ed. (North-Holland, Amsterdam, 1971) pp. 235–307.
[20] S. Stenlund, *Combinators, λ-terms and proof theory* (D. Reidel Publ. Co., Dordrecht, 1972).
[21] W. W. Tait, Intensional interpretations of functionals of finite type I, *J. Symbolic Logic* 32 (1967) 198–212.
[22] A. S. Troelstra, Notions of realizability for intuitionistic arithmetic and intuitionistic arithmetic in all finite types, in: *Proceedings of the second Scandinavian logic Symposium*, J. E. Fenstad, Ed. (North-Holland, Amsterdam, 1971) pp. 369–405.