STUDIES IN LOGIC

AND

THE FOUNDATIONS OF MATHEMATICS

VOLUME 80

NORTH-HOLLAND PUBLISHING COMPANY–AMSTERDAM · OXFORD
AMERICAN ELSEVIER PUBLISHING COMPANY, INC.–NEW YORK

LOGIC COLLOQUIUM '73

PROCEEDINGS OF THE LOGIC COLLOQUIUM
BRISTOL, JULY 1973

Edited by

H. E. ROSE

Lecturer in Mathematics, Bristol

and

J. C. SHEPHERDSON

Professor of Pure Mathematics, Bristol

1975

NORTH-HOLLAND PUBLISHING COMPANY-AMSTERDAM · OXFORD
AMERICAN ELSEVIER PUBLISHING COMPANY, INC.-NEW YORK

CONTENTS

INVITED PAPERS

CONTRIBUTED PAPERS

ELIMINATION OF HIGHER TYPE LEVELS IN DEFINITIONS OF PRIMITIVE RECURSIVE FUNCTIONALS BY MEANS OF TRANSFINITE RECURSION

Helmut SCHWICHTENBERG[1]
Münster, Westfalen, W. Germany

0. Introduction

Hilbert's program asks for finitary consistency proofs for formalized mathematical theories. This program appears to be even more natural, if one extends it a little (Kreisel [9]): Given a formalized version of an abstract proof for a finitary assertion (example: proof of $x + y = y + x$, x, y variables for natural numbers, in an axiomatic set theory), can one always construct from this a finitary proof of the same assertion? By a well-known result of Gödel, this is impossible if it is required that the finitary methods used should be formalizable in the abstract theory. However, a way to overcome this difficulty was already proposed by Hilbert (in the Introduction to [6, Volume I]): 'Jenes Ergebnis zeigt ... nur, daß man für die weitergehenden Widerspruchsfreiheitsbeweise den finiten Standpunkt in einer schärferen Weise ausnutzen muß als dies bei der Betrachtung der elementaren Formalismen erforderlich ist'. The development and investigation of strong, but still finitary methods is therefore a central theme in proof theory.

For first order number theory Z, Hilbert's program was carried out first by Gentzen [3] (see also Gentzen [4]). Gentzen assigns to every formal proof p in Z an ordinal $\alpha_p < \epsilon_0$ and defines a reduction procedure for proofs in Z such that any reduction step preserves the end formula, but lowers the assigned ordinal. Since proofs in Z with assigned ordinal o

[1] I want to thank Professor S. Feferman, Professor W. Howard and Professor G. Kreisel for their advice and valuable suggestions.

certainly do not prove a contradiction, one can conclude by transfinite induction up to α_p that p does not prove a contradiction. All this can be formalized in the quantifier-free primitive recursive (pr rec) arithmetic PRA extended by transfinite induction up to arbitrary ordinals $< \epsilon_0$, and also (Kreisel [10]) in PRA extended by definition schemata for α-recursion, $\alpha < \epsilon_0$. Hence the consistency of the latter theory, $\mathrm{PRA}_{<\epsilon_0}$ say, implies the consistency of Z. Conversely, the consistency of Z implies the consistency of $\mathrm{PRA}_{<\epsilon_0}$, for $\mathrm{PRA}_{<\epsilon_0}$ is a subtheory of the conservative (by Hilbert–Bernays [6]) extension of Z by α-recursion, $\alpha < \epsilon_0$.

Another consistency proof for Z is due to Gödel [5], in a paper entitled 'Über eine bisher noch nicht benützte Erweiterung des finiten Standpunkts': Gödel shows, that Z can be interpreted in a quantifier-free extension T of PRA to functionals of finite types. Hence the consistency of T implies the consistency of Z. Again the converse holds (Kreisel [11]).

So the proof-theoretic strength of Z is expressed in the quantifier-free theories $\mathrm{PRA}_{<\epsilon_0}$ and T in different ways by definition schemata: In $\mathrm{PRA}_{<\epsilon_0}$ by allowing α-recursion, $\alpha < \epsilon_0$, for defining functions, and in T by extending the schemata of explicit definition and primitive recursion available in PRA to functionals of finite types. Hence there is an obvious question how to compare more directly and in a general form these two methods of extending simple definition schemata.

That any function, definable in $\mathrm{PRA}_{<\epsilon_0}$, is definable in T was proved first by Kreisel [12, §3.4], using Gödel [5]. More generally, Tait [21] shows that any 2^α-recursion can be reduced to an α-recursion with a type level greater by one. In the other direction only a special case is treated in the literature, namely that any function definable in T is definable in $\mathrm{PRA}_{<\epsilon_0}$ too. In Tait [19] it is mentioned that this follows from Kreisel [11] (with Gentzen [4]). In the same paper, Tait sketches a more direct proof[2]. Here we show in a general form (e.g. for functionals instead of functions) that 'detours' through higher type levels can be eliminated by means of transfinite recursion.

We obtain the following result (for a precise formulation, see Section 3.8): Let a functional F of type level $n+1$ be defined by explicit definitions and α-recursions. All auxiliary functionals introduced by recursion shall have type levels $\leq n+m+1$ $(m \geq 1)$. Then one can find a new definition of F, containing auxiliary functionals of type level $\leq n+1$

[2] For other proofs, see Howard [7] and Parsons [15].

only, but using a β-recursion with $\beta < 2_m(\alpha\omega)$ instead of the α-recursions (where $2_0(\xi) := \xi$, $2_{i+1}(\xi) := 2^{2_i(\xi)}$).

In fact, we obtain a stronger formal version of this result: Let T_α be the theory obtained from Gödel's T (with weak extensionality, see Spector[18])[3] by adding α-recursion. Let T_α^n be the subtheory of T_α obtained by restricting all constants to those of type level $\leq n+1$. Then for every constant F in T_α^{n+m} $(m \geq 1)$ of type level $n+1$ one can find a constant F' of T_β^n with $\beta < 2_m(\alpha\omega)$ such that $Fx_1 \dots x_k = F'x_1 \dots x_k$ is provable in T_β^{n+m}.

The proof runs as follows. First, Tait has observed (in [19]) that any α-recursive functional F can be represented by an infinite term t_F, i.e., a term built up from typed variables and numerals by application, abstraction and the formation of sequences $\langle t_i \rangle_{i \in \mathbf{N}}$ of type $0 \to 0$ with terms t_i of type 0. For instance, if F of type $0 \to \tau$ is defined by a $<$-recursion

$$F(x) = G([F]_{<x}, x)$$

($<$ well-ordering of $\mathbf{N}$, $[F]_{<x}$ course of values of F below x, i.e., $[F]_{<x}(y) := F(y)$ if $y < x$, and $:= \mathbf{0}$ otherwise), and G is already represented by an infinite term t_G, then one can represent $F(x)$ by

$$t_x :\equiv t_G \langle t_{xi} \rangle_{i \in \mathbf{N}} \bar{x},$$

with

$$t_{xi} :\equiv \begin{cases} t_i & \text{if } i < x, \\ \mathbf{0} & \text{otherwise,} \end{cases}$$

where $\mathbf{0} :\equiv \lambda x_1 \dots x_k . 0$. So F can be represented by $\langle t_x \rangle_{x \in \mathbf{N}}$. Here a sequence $\langle s_i \rangle_{i \in \mathbf{N}}$ of terms s_i of type $\sigma \neq 0$ is an abbreviation for $\lambda x x_1 \dots x_k . \langle s_i x_1 \dots x_k \rangle_{i \in \mathbf{N}} x$[4]. One can see easily that for any α-recursive functional F the depth $|t_F|$ (defined as usual) of the representing term t_F is $< \alpha\omega$.

Let the rank $R(t)$ of an infinite term t be the supremum of the type levels of all subterms λxs in a context $(\lambda xs)r$. Now let F be an α-recursive functional of type level $n+1$. In the definition of F, all the auxiliary functionals shall have type levels $\leq n+m+1$. Clearly one can assume that the infinite term t_F representing F has a rank $R(t_F) \leq$

[3] I do not know whether a similar result holds without extensionality.

[4] I am grateful to Mr. R. Statman for telling me this possibility of eliminating sequences of terms of a type $\neq 0$ in an extensional context.

$n + m + 1$. We will define a reduction relation (essentially as in Tait[19], but using λ-conversions $(\lambda xs)r \to s_x[r]$ only) such that every infinite term t of rank $R(t) \le k + 1$ can be reduced to a t' with $R(t') \le k$ and $|t'| \le 2^{|t|}$. Hence, the above functional F of type level $n + 1$ can be represented by an infinite term t_F^* with rank $R(t_F^*) \le n + 1$ and depth $|t_F^*| < 2_m(\alpha\omega)$.

We consider now finite notations or numbers for infinite terms. To construct them, there is no problem for variables, numerals, application and abstraction. In the case of a sequence $\langle t_i \rangle_{i \in \mathbf{N}}$, the number contains (among other things)

(i) an index e of a pr rec function that gives, when applied to i, a number for t_i,

(ii) a bound for the depth of $\langle t_i \rangle_i$.

Next we define valuation functionals $\mathrm{Val}_\tau^{\alpha, M}$ with the following property. Let $\ulcorner t \urcorner$ be a number of a closed infinite term t of type τ and depth $|t| < \alpha$, all of whose subterms have types from a finite set M. Then $\mathrm{Val}_\tau^{\alpha, M} \ulcorner t \urcorner$ represents the same set theoretic functional as t. The definition of the $\mathrm{Val}_\tau^{\alpha, M}$, $\tau \in M$, is by simultaneous α-recursion[5].

Now let F be as before. We first obtain a number $\ulcorner t_F \urcorner$ of an infinite term t_F representing F, such that $F = \mathrm{Val}_\tau^{\alpha k, M} \ulcorner t_F \urcorner$ for suitable k, M. Corresponding to the reduction of the term t_F with $R(t_F) \le n + m + 1$ and $|t_F| < \alpha\omega$ to a term t_F^* with $R(t_F^*) \le n + 1$ and $|t_F^*| < 2_m(\alpha\omega)$ we construct a function Red* such that

$$F = \mathrm{Val}_\tau^{\alpha k, M} \ulcorner t_F \urcorner = \mathrm{Val}_\tau^{\beta, M_{n+1}} (\mathrm{Red}^* \ulcorner t_F \urcorner)$$

with a $\beta < 2_m(\alpha\omega)$ and a set M_{n+1} of types with level $\le n + 1$. The function Red* turns out to be primitive recursive[6]. Since $\mathrm{Val}_\tau^{\beta, M_{n+1}}$ is obtained using β-recursion but without auxiliary functionals of type level $> n + 1$, we have the desired result.

To get the formal version mentioned above, one has to formalize this proof in a T_β^{n+m}, $\beta < 2_m(\alpha\omega)$. The necessary modifications in the informal proof are discussed in Section 4. Specifically, the predicate 'u is a number of a term' is Π_1^0. Hence, a theorem with this predicate in the premiss and proved by induction cannot immediately be formalized in a T_β^{n+m}. To

[5] With some more work one can define analogous valuation functionals, where instead of a finite set M of types the infinite set of all types of level $\le n$ occurs.

[6] A similar situation occurs in the theory of Kleene's 0, where $+_0$ can be chosen pr rec; cf. Kleene[8].

overcome this difficulty we use Herbrand's Theorem (cf. Kreisel [9] and Shepherdson [17]).

1. Functionals defined by transfinite recursion

1.1. Types are 0 and with σ, τ also $(\sigma \to \tau)$. The classes $\mathcal{F}_\tau$ of all (set theoretic) *functionals* of type τ are defined by $\mathcal{F}_0 := \mathbf{N}$, $\mathcal{F}_{\sigma\to\tau} := \mathcal{F}_\tau^{\mathcal{F}_\sigma} := \{F^{\sigma\to\tau} : F^{\sigma\to\tau} : \mathcal{F}_\sigma \to \mathcal{F}_\tau\}$. Many-place functions can be reduced as usual to 1-place functions; e.g., a 2-place function can be considered as a functional of type $0 \to (0 \to 0)$. We write $\tau_1 \to \tau_2 \to \cdots \tau_{n-1} \to \tau_n$ for $\tau_1 \to (\tau_2 \to \cdots (\tau_{n-1} \to \tau_n) \cdots)$. Any type τ can be written uniquely in the form $\tau = \tau_1 \to \tau_2 \to \cdots \tau_n \to 0$, as one proves easily by induction on τ; we call $\tau_1, \ldots, \tau_n$ the *argument types* of τ. We write $F^{\tau_1\to\cdots\tau_n\to\tau}(G_1^{\tau_1}, \ldots, G_n^{\tau_n})$ for $F^{\tau_1\to\cdots\tau_n\to\tau}(G_1^{\tau_1}) \ldots (G_n^{\tau_n})$. Type indices clear from the context will be omitted frequently. The *type level* $L(\tau)$ of a type τ is defined by $L(0) := 0$, $L(\sigma \to \tau) := \max(L(\sigma) + 1, L(\tau))$.

1.2. DEFINITION. The class of *primitive recursive functionals* is the smallest class

(i) containing the number 0^0 and the successor function $S^{0\to 0}$,

(ii) closed under explicit definitions

$$F^{\tau_1\to\cdots\tau_n\to 0}(x_1^{\tau_1}, \ldots, x_n^{\tau_n}) = A$$

with A built up from the variables $x_1, \ldots, x_n$ and already defined functionals $G_1, \ldots, G_m$ by application, and

(iii) closed under primitive recursion

$$F(0) = G^\tau$$
$$F(x+1) = H(F(x), x).$$

1.3. We want to consider transfinite recursion too. Let $<$ be an arbitrary well-ordering of $\mathbf{N}$.

DEFINITION. The class of *$<$-recursive functionals* is the smallest class with the properties (i)–(iii) and

(iv) closed under $<$-recursion

$$F(x) = G([F]_{<x}, x),$$

where $[F]_{<x}$ is the course-of-values of F below x, i.e., $[F]_{<x}(y) := F(y)$ if $y < x$ and $:= \mathbf{0}$ otherwise.

1.4. REMARK. It is well known that within the class of recursive functions, the strength of the schema of $<$-recursion is determined neither by the order type nor by the recursion theoretic complexity of the relation $<$. Myhill [14] and Routledge [16] prove that any recursive function can be defined by pr rec operations and just one $<$-recursion with a pr rec well-ordering $<$ of order type ω. A dependence of the strength of $<$-recursion on (and only on) the order type of $<$ can be obtained, if one restricts oneself to *standard well-orderings* $<$. However, this notion is available only for concrete order types such as ϵ_0, Γ_0. If $<$ is a standard well-ordering of order type α, $<$-recursion is also called α-recursion.

1.5. REMARK. For simplicity, we consider only functionals with definition trees containing transfinite recursion with respect to only one well-ordering $<$. The general case can be handled correspondingly.

1.6. We want to simplify a little the definition of $<$-recursive functionals. For this we show that under simple assumptions on $<$ the schema of primitive recursion

$$F(0) = G,$$
$$F(x+1) = H(F(x), x)$$

can be reduced to the schema of $<$-recursion. So let G, H be given. We want to define a solution F to the above equations by means of $<$-recursion. Assume that there exist pr rec functions h, h' such that

$$x < y \rightarrow h(x) < h(y), \tag{1}$$

$$h'(h(x)) = x. \tag{2}$$

Let

$$F(x) = F_1(h(x)),$$
$$F_1(y) = G_1([F_1]_{<y}, y),$$
$$G_1(z, y) = \begin{cases} G & \text{if } y = h(0), \\ H(z(h(x)), x) \text{ with } x := h'(y) - 1 & \text{otherwise.} \end{cases}$$

One can check easily that F has the required properties.

Hence, under the assumptions (1), (2) (which we will presuppose whenever dealing with $<$-recursive functionals) the definition of $<$-recursive functionals given above is equivalent to the following: The class of $<$-recursive functionals is the smallest class containing the number 0 and all pr rec functions and closed under explicit definition (ii) and $<$-recursion (iv).

2. Infinite terms

We define infinite terms and construct in a canonical way for any definition of a $<$-recursive functional F an infinite term t_F representing F. Furthermore, we define a procedure by means of which t_F can be reduced to a term t^*_F all of whose subterms have type levels not greater than the type level of the whole term.

The proofs are to a large extent parallel to Tait [19]. We carry them out, since on the one hand some changes are useful, and on the other hand some details of the construction will be referred to later.

2.1. For any type τ, let countably many τ-variables (i.e., variables of type τ) x^τ, y^τ, z^τ, ... be given.

Definition of (*infinite*) τ-*terms*

(i) Any τ-variable x^τ is a τ-term.

(ii) For any natural number $n \geq 0$, $\bar{n}$ is a 0-term.

(iii) If t is a $(\sigma \to \tau)$-term and s is a σ-term, then (ts) is a τ-term.

(iv) If t is a τ-term, then $\lambda x^\sigma.t$ is a $(\sigma \to \tau)$-term.

(v) If for each $i \in \mathbf{N}$, t_i is a 0-term, then the sequence $\langle t_i \rangle_{i \in \mathbf{N}}$ is a $(0 \to 0)$-term.

(vi) t is a τ-term only as required by (i)–(v).

Infinite terms are denoted by t, s, r. The type level $L(\tau)$ of a τ-term t is denoted by $L(t)$. For (ts) we also write $t(s)$ and for $t(s_1)\dots(s_n)$ also $t(s_1, \dots, s_n)$ or $ts_1 \dots s_n$. For $\lambda x_1 . \lambda x_2 . \dots \lambda x_n . t$ we write $\lambda x_1 \dots x_n . t$. The terms $\bar{n}$ are called *numerals*.

Ultimately we are interested in closed terms only, but we have to consider subterms of them. Now, as one can see easily from the definition, subterms of closed infinite terms have only a finite number of free variables. So it is sufficient to consider infinite terms with only a finite number of free variables, and we will do that from now on.

2.2. For any infinite term t one can define its value $\mathrm{Val}^{\mathfrak{a}}_{\mathfrak{x}} t$ under an assignment of the functionals $\mathfrak{a} = a_1, \dots, a_n$ to the variables $\mathfrak{x} = x_1, \dots, x_n$, if all the free variables of t occur among $\mathfrak{x}$. $\mathrm{Val}^{\mathfrak{a}}_{\mathfrak{x}} t$ is a functional of type τ, if t is a τ-term.

DEFINITION. $\mathrm{Val}_{\xi}^{a}\, t$ is defined by

(i) $\mathrm{Val}_{\xi}^{a} x_i = a_i$,
(ii) $\mathrm{Val}_{\xi}^{a} \bar{n} = n$,
(iii) $\mathrm{Val}_{\xi}^{a}(ts) = (\mathrm{Val}_{\xi}^{a} t)(\mathrm{Val}_{\xi}^{a} s)$,
(iv) $(\mathrm{Val}_{\xi}^{a}(\lambda xt))a = \mathrm{Val}_{\xi x}^{aa} t$,
(v) $(\mathrm{Val}_{\xi}^{a}\langle t_i\rangle_i)n = \mathrm{Val}_{\xi}^{a} t_n$.

2.3. DEFINITION. $|t|$ (the *depth* of t) is defined by

(i) $|x| = 1$,
(ii) $|\bar{n}| = 1$,
(iii) $|ts| = \max(|t|, |s|) + 1$,
(iv) $|\lambda xt| = |t| + 1$,
(v) $|\langle t_i\rangle_i| = \sup_i(|t_i| + 1)$.

2.4. For any $<$-recursive functional F we construct a closed infinite term t representing F (i.e., $F = \mathrm{Val}(t)$) with depth $|t| < \alpha\omega$, α order type of $<$.

0 is represented by $\bar{0}$ and a say 2-place pr rec function f by $\lambda xy\langle\langle\overline{f(i,j)}\rangle_j y\rangle_i x$. Let F be defined explicitly from functionals $G_1, \ldots, G_m$ in the form $F(x_1, \ldots, x_n) = A$. Assume that terms $s_1, \ldots, s_m$ representing $G_1, \ldots, G_m$ and with $|s_i| < \alpha\omega$ are already constructed. Let t be built up from $s_1, \ldots, s_m$ by applications in the same way A is built up from $G_1, \ldots, G_m$. Then F is represented by $\lambda x_1 \ldots x_n . t$ and we have $|\lambda x_1 \ldots x_n . t| < \alpha\omega$. Let finally F be introduced by a $<$-recursion from G:

$$F(x) = G([F]_{<x}, x).$$

Assume that a term s representing G with $|s| < \alpha\omega$ is already constructed. Then one can define recursively representations t_n of $F(n)$ by

$$t_n :\equiv s\langle t_{ni}\rangle_i \bar{n},$$

with

$$t_{ni} :\equiv \begin{cases} t_i & \text{if } i < n, \\ \mathbf{0} & \text{otherwise} \end{cases}$$

and represent F by $t :\equiv \langle t_n\rangle_n$[7]. $|t| < \alpha\omega$ can be seen as follows. Let $|s| < \alpha k$. By $<$-induction on n one shows easily $|t_n| \leq \alpha k + l(o_<(n) + 1)$ with $l < \omega$, $o_<(n)$ order type of $\{m : m < n\}$. From this one obtains immediately $|t| < \alpha\omega$ (if α is a limit number, one has $|t| < \alpha(k+1)$).

[7] For terms $\langle s_i^{\sigma}\rangle_i$ with $\sigma \neq 0$, cf. the introduction and footnote 3.

2.5. For a given $<$-recursive functional F of type level $n+1$, the term t representing F constructed above contains in general subterms of a type level $>n+1$; clearly the supremum of all type levels of subterms equals the maximum type level of an auxiliary functional in the definition of F. We show now that one can also choose a term representing F, whose 'inner type level' depends only on the auxiliary functionals introduced by recursion.

LEMMA. *Let a definition of a $<$-recursive functional F of type level $n+1$ be given. All auxiliary functionals introduced by recursion shall have type levels $\leq n+m+1$. Then one can construct a closed infinite term t_F with $F = \mathrm{Val}(t_F)$ and $|t_F| < \alpha\omega$ (α order type of $<$), such that all subterms of t_F have type levels $\leq n+m+1$.*

PROOF. The proof is obtained by a simple modification of the construction in Section 2.4. We use the normal form theorem for finite terms (i.e., terms built up without using sequences $\langle t_i \rangle_i$): Any finite term A can be reduced by λ-conversions $(\lambda x B)C \to B_x[C]$ of subterms to a finite term A^* of rank $RA^* = 0$ (a proof of this can be obtained e.g. by specializing Section 2.10). The cases $F = 0$ and F pr rec function are trivial. Let F be defined explicitly in the form $F(x_1, \ldots, x_n) = A[x_1, \ldots, x_n, G_1, \ldots, G_k]$. We can assume that $G_1, \ldots, G_k$ are introduced by $<$-recursion, if we allow A to contain abstractions. If A^* is the normal form of A, then $t_F :\equiv \lambda x_1 \ldots x_n \,.\, A^*[x_1, \ldots, x_n, t_{G_1}, \ldots, t_{G_k}]$ has the required properties. Let finally F be defined by a $<$-recursion $F(x) = A[[F]_{<x}, x, G_1, \ldots, G_p]$ with a finite term A and $G_1, \ldots, G_p$ introduced by $<$-recursion. Let A^* be the normal form of A. We define terms t_n representing $F(n)$ by

$$t_n :\equiv A^*[\langle t_{ni} \rangle_i, \bar{n}, t_{G_1}, \ldots, t_{G_p}],$$

with

$$t_{ni} :\equiv \begin{cases} t_i & \text{if } i < n, \\ \mathbf{0} & \text{otherwise} \end{cases}$$

and let again $t_F :\equiv \langle t_n \rangle_n$. Clearly t_F has the required properties ($|t_F| < \alpha\omega$ is proved just as in Section 2.4; cf. also Section 3.3).

2.6. As usual one can define substitution $t_x[s]$. By induction on t one proves easily

$$\mathrm{Val}^{\mathfrak{a}}_{g} t_y[s] = \mathrm{Val}^{\mathfrak{a}\,\mathrm{Val}^{\mathfrak{a}}_{g} s}_{g\;\;y}\, t,$$

$$|t_x[s]| \leq |s| + |t|.$$

2.7. DEFINITION. $t \dashv s$ (t *reduces* to s) is defined by

(i) $(\lambda xt)s \dashv t_x[s]$

(ii) $\dashv$ *is reflexive and transitive.*

(iii) If $t \dashv t'$ and $s \dashv s'$, then $ts \dashv t's'$.

(iv) If $t \dashv t'$, then $\lambda xt \dashv \lambda xt'$.

(v) If $t_i \dashv t'_i$ for each i, then $\langle t_i \rangle_i \dashv \langle t'_i \rangle_i$.

(vi) $t \dashv s$ only as required by (i)–(v).

Obviously, reducing a term does not change its value, i.e., if $t \dashv s$, then $\mathrm{Val}^\alpha_g t = \mathrm{Val}^\alpha_g s$.

2.8. The *rank* Rt of a term t is defined as the supremum of the type levels of all subterms of the form λxs in a context $(\lambda xs)r$.

2.9. LEMMA. $Rt_x[s] \le \max(Rt, Rs, Ls)$.

PROOF. The proof by induction on t is straightforward.

2.10. LEMMA. *If $Rt \le k+1$, then there is a t' such that $t \dashv t'$, $Rt' \le k$ and $|t'| \le 2^{|t|}$.*

PROOF. The proof is by induction on t.

Case 1: $t \equiv \langle t_i \rangle_i$. Take $t' \equiv \langle t'_i \rangle_i$ with t'_i chosen by ind. hyp.

Case 2: $t \equiv \lambda xs$. Take $t' \equiv \lambda xs'$.

Case 3: $t \equiv rs$. Choose r', s' by ind.hyp. If r' has the form λxr_1 and $Lr = k+1$, take $t' \equiv (r_1)_x[s']$; it follows $t \dashv t'$, $Rt' \le \max(Rr_1, Rs', Ls') \le k$ and

$$\begin{aligned} |t'| &\le |s'| + |r'| \\ &\le 2^{|s|} + 2^{|r|} \\ &\le 2^{\max(|s|,|r|)+1} \\ &= 2^{|t|}. \end{aligned}$$

Otherwise, take $t' \equiv r's'$; it follows $t \dashv t'$, $|t'| \le \max(2^{|r|}, 2^{|s|}) + 1 \le 2^{|t|}$ and with a simple argument $Rt' \le k$.

2.11. REMARK. This construction of t' from t contains substitutions and hence some changes of bound variables may be necessary. I do not know whether one can arrange matters in such a way that the number of bound variables in t' will be finite if it was finite in t. If this is the case, then we could use in Section 3 a simpler version of the valuation functionals.

2.12. Consider now a definition of a $<$-recursive functional F of type level $n+1$. All auxiliary functionals introduced by recursion shall have type levels $\leq n+m+1$. By Section 2.5 we can write F in the form $F = \mathrm{Val}(t_F)$ with a t_F of depth $|t_F| < |{<}|\omega$ and of rank $Rt_F \leq n+m+1$. From t_F, by m-fold application of Section 2.10 we obtain a t_F^* with $t_F \dashv t_F^*$, $Rt_F^* \leq n+1$ and $|t_F^*| < 2_m(|{<}|\omega)$, and we also have $F = \mathrm{Val}(t_F^*)$. Our aim in the next section will be to find corresponding to the representation $\mathrm{Val}(t_F^*)$ of F a definition of F, which does not make 'detours' through higher type levels, but uses instead a recursion with a well-ordering of order type $\beta < 2_m(|{<}|\omega)$, constructed canonically from $<$.

3. Term numbers and valuation functionals

3.1. We define numbers or notations for certain infinite terms, similar to the ordinal notations of Church and Kleene. The definition is trivial in the case of variables, numerals, application or abstraction. In the case of a sequence we use a pr rec function for enumerating the members of the sequence; hence for evaluating a term number we need only recursive functions of a bounded complexity. Our definition of term numbers contains the following parameters:

(1) a relation $<$ to be used for giving bounds of the depth of the denoted terms. The evaluation of a term number can then be done by $<$-recursion.

(2) a set M of types.

We assume that an indexing of the pr rec functions is given, as in Kleene[8] say. The pr rec function with index e is denoted by $[e]$. Let $<$ be a 2-place relation and $a_1 \in \mathrm{Field}(<)$ (in a well-ordering $<$, a_1 is to be the element corresponding to the ordinal 1). Let M be a set of types. All types occurring in the following definition shall be from M.

DEFINITION. $u \in \mathrm{Num} = \mathrm{Num}^{<,M}$ (u is *term number*) is defined by

(i) $\langle 1, \ulcorner\tau\urcorner, a_1, i\rangle =: \ulcorner x_i^\tau \urcorner \in \mathrm{Num}$.

(ii) $\langle 2, \ulcorner 0\urcorner, a_1, i\rangle =: \ulcorner \bar{\imath}\urcorner \in \mathrm{Num}$.

(iii) If $u, v \in \mathrm{Num}$, $\mathrm{Type}(u) = \ulcorner\sigma \to \tau\urcorner$, $\mathrm{Type}(v) = \ulcorner\sigma\urcorner$ and $|u|, |v| < a$, then $\langle 3, \ulcorner\tau\urcorner, a, u, v\rangle \in \mathrm{Num}$.

(iv) If $u \in \mathrm{Num}$, $\mathrm{Type}(u) = \ulcorner\tau\urcorner$ and $|u| < a$, then $\langle 4, \ulcorner\sigma\to\tau\urcorner, a, \ulcorner x_i^\sigma\urcorner, u\rangle \in \mathrm{Num}$.

(v) If for each i $[e](i) =: u_i \in \mathrm{Num}$, $\mathrm{Type}(u_i) = \ulcorner 0\urcorner$, $|u_i| < a$, $R(u_i) \leq k$ and $\mathrm{FV}(u_i) \subseteq^* b$, then $\langle 5, \ulcorner 0 \to 0\urcorner, a, k, b, e\rangle \in \mathrm{Num}$.

(vi) $u \in \mathrm{Num}$ only as required by (i)–(v).

Here $\text{Type}(u) := (u)_1$, $|u| := (u)_2$ and R, FV (the notations for rank and free variables) are defined by

$$R(u) = \begin{cases} 0 & \text{if } (u)_0 = 1, 2, \\ \max(R((u)_3), R((u)_4)) & \text{if } (u)_0 = 3, (u)_{3,0} \neq 4, \\ \max(R((u)_3), R((u)_4), L((u)_3)) & \text{if } (u)_0 = 3, (u)_{3,0} = 4, \\ R((u)_4) & \text{if } (u)_0 = 4, \\ (u)_3 & \text{otherwise.} \end{cases}$$

$$\text{FV}(u) = \begin{cases} \{u\}^{\#} & \text{if } (u)_0 = 1, \\ \emptyset^{\#} & \text{if } (u)_0 = 2, \\ \text{FV}((u)_3) \cup^{\#} \text{FV}((u)_4) & \text{if } (u)_0 = 3, \\ \text{FV}((u)_4) -^{\#} \{(u)_3\}^{\#} & \text{if } (u)_0 = 4, \\ (u)_4 & \text{otherwise,} \end{cases}$$

where the set-theoretic symbols augmented by $\#$ shall correspond under a (trivial) coding of finite sets of variables to the set theoretic operations denoted by the same symbol. $\ulcorner \tau \urcorner$ denotes as usual a Gödel number of τ, and $L(u)$ reads off from u the level of the type τ with $(u)_1 = \ulcorner \tau \urcorner$.

3.2. Any term number u determines uniquely an infinite term t_u, as follows.

(i) If $u = \langle 1, \ulcorner \tau \urcorner, a_1, i \rangle \in \text{Num}$, then $t_u \equiv x_i^\tau$.

(ii) If $u = \langle 2, \ulcorner 0 \urcorner, a_1, i \rangle \in \text{Num}$, then $t_u \equiv \bar{i}$.

(iii) If $u = \langle 3, \ulcorner \tau \urcorner, a, v, w \rangle \in \text{Num}$, then $t_u = t_v t_w$.

(iv) If $u = \langle 4, \ulcorner \sigma \to \tau \urcorner, a, \ulcorner x_i^\sigma \urcorner, v \rangle \in \text{Num}$, then $t_u \equiv \lambda x_i^\sigma t_v$.

(v) If $u = \langle 5, \ulcorner 0 \to 0 \urcorner, a, k, b, e \rangle \in \text{Num}$, then $t_u \equiv \langle t_{u_i} \rangle_i$ with $u_i := [e](i)$.

Conversely, a number of an infinite term does not always exist, and if it exists, it is in general not uniquely determined. We use $\ulcorner t \urcorner$ as a variable for numbers of t, and by using $\ulcorner t \urcorner$ in a given context we presuppose that a number of t exists.

3.3. For any $<$-recursive functional F, an infinite term t_F representing F was constructed in Section 2.5. Now we define a number $\ulcorner t_F \urcorner$ for t_F in a canonical way.

Let $<$ be the well-ordering of $\mathbf{N}$ used in the definition of the $<$-recursive functional F. Let α be the order type of $<$. From $<$ one obtains canonically a well-ordering $<_0$ of order type $\alpha\omega$ (for instance by $n <_0 m :\leftrightarrow (\pi_1 n < \pi_1 m \wedge \pi_2 n = \pi_2 m) \vee \pi_2 n < \pi_2 m$; π_1, π_2 are pr rec inverse functions for a pr rec surjective pairing function π). We assume that $<$ and the number theoretic functions corresponding via $<$ to the ordinal functions $\lambda\xi.\xi+1$, $\lambda\xi.n\cdot\xi$ are pr rec. Let M be the set of all those types occurring in the definition of F that have a level $\leq$ the maximum type level l of an auxiliary functional introduced by $<$-recursion.

We define $\ulcorner t_F \urcorner \in \mathrm{Num}^{<_0, M}$ by induction on F, confining ourselves to the case of an F introduced by $<$-recursion. The other cases are simpler or trivial.

So let F be defined in the form $F(x) = A[[F]_{<x}, x, G_1, \ldots, G_p]$ with $G_1, \ldots, G_p$ introduced by $<$-recursion. Let A^* be the normal form of A. t_F was defined from terms

$$t_n \equiv A^*[\langle t_{ni}\rangle_i, \bar{n}, t_{G_1}, \ldots, t_{G_p}]$$

with

$$t_{ni} \equiv \begin{cases} t_n & \text{if } i < n, \\ \mathbf{0} & \text{otherwise} \end{cases}$$

by $t_F \equiv \langle t_n\rangle_n$. $|t_F| < \alpha\omega$ was obtained as follows: Assume $|t_{G_1}|, \ldots, |t_{G_p}| \leq \alpha k$ and $|\mathbf{0}| + 1 \leq k$. Then $|t_n| \leq \alpha k + (|A^*| + 1)(o_<(n) + 1)$ (one proves this and $|\langle t_{ni}\rangle_i| \leq \alpha k + (|A^*| + 1)o_<(n) + 1$ together by induction on n ; $o_<(n)$ is the ordinal corresponding in $<$ to the number n).

Assume that term numbers $\ulcorner t_{G_1}\urcorner, \ldots, \ulcorner t_{G_p}\urcorner$ for $G_1, \ldots, G_p$ with depth bounds $|\ulcorner t_{G_i}\urcorner| \leqslant_0 \ulcorner \alpha k\urcorner$ are already constructed. By means of the recursion theorem for the class $\mathfrak{P}$ of pr rec functions (Kleene[8, p. 75]) we define a pr rec function with index e such that $[e](n) =: u_n$ is a term number for t_n with a depth bound $|u_n| \leqslant_0 \ulcorner \alpha k + (|A^*| + 1)(o_<(n) + 1)\urcorner$ (because of the assumptions on $<$, $\ulcorner \ldots n \ldots \urcorner$ is a pr rec function of n; $\ulcorner \ldots n \ldots \urcorner$ means of course the number corresponding in $<_0$ to the ordinal $\ldots n \ldots$). Assume that such an e is already constructed. Then one obtains a number u'_n for $\langle t_{ni}\rangle_i$ in the form $\langle 5, \ulcorner 0 \to 0\urcorner, a, 1, \emptyset^*, e'\rangle$ with $e' = e'(e, n)$ such that

$$[e'](i) = \begin{cases} [e](i) & \text{if } i < n, \\ \ulcorner \mathbf{0} \urcorner & \text{otherwise,} \end{cases}$$

and $a = a(n) = \ulcorner \alpha k + (|A^*| + 1)o_<(n) + 1\urcorner$. $e'(e, n)$ and $a(n)$ are pr rec. Corresponding to the way t_n is built up from $\langle t_{ni}\rangle_i, t_{G_1}, \ldots, t_{G_p}$ one can construct from $u'_n, \ulcorner t_{G_1}\urcorner, \ldots, \ulcorner t_{G_p}\urcorner$ primitive recursively a number for t_n, with a depth bound $\leqslant_0 \ulcorner \alpha k + (|A^*| + 1)(o_<(n) + 1) \urcorner$. An application of the recursion theorem for $\mathfrak{P}$ now gives the required e. From e one obtains immediately a number $\ulcorner t_F \urcorner$ for t_F.

3.4. We want to define now valuation functionals $\mathrm{Val}_\tau = \mathrm{Val}_\tau^{<,M}$, $\tau \in M$. Let u be a term number $\in \mathrm{Num}^{<,M}$ with $\mathrm{Type}(u) = \ulcorner \tau \urcorner$ and $\mathrm{FV}(u) = \emptyset^*$. We would like to have $\mathrm{Val}_\tau(u)$ giving the value of the closed term denoted by u. However, for the recursive definition of the Val_τ it is necessary to allow free variables in t_u. Therefore, we introduce an additional argument which is supposed to code an assignment $\mathfrak{x} \to \mathfrak{a}$ for these variables. So we want to have (type indices are omitted)

$$\mathrm{Val}\ulcorner x_i \urcorner \ulcorner \mathfrak{x} \to \mathfrak{a} \urcorner = a_i,$$
$$\mathrm{Val}\ulcorner \bar{i} \urcorner \ulcorner \mathfrak{x} \to \mathfrak{a} \urcorner = i,$$
$$\mathrm{Val}\ulcorner ts \urcorner \ulcorner \mathfrak{x} \to \mathfrak{a} \urcorner = (\mathrm{Val}\ulcorner t \urcorner \ulcorner \mathfrak{x} \to \mathfrak{a} \urcorner)(\mathrm{Val}\ulcorner s \urcorner \ulcorner \mathfrak{x} \to \mathfrak{a} \urcorner),$$
$$\mathrm{Val}\ulcorner \lambda x t \urcorner \ulcorner \mathfrak{x} \to \mathfrak{a} \urcorner a = \mathrm{Val}\ulcorner t \urcorner \ulcorner \mathfrak{x}, x \to \mathfrak{a}, a \urcorner,$$
$$\mathrm{Val}\ulcorner \langle t_i \rangle_i \urcorner \ulcorner \mathfrak{x} \to \mathfrak{a} \urcorner n = \mathrm{Val}([e](n)) \ulcorner \mathfrak{x} \to \mathfrak{a} \urcorner,$$

where e is the pr rec index of the sequence of the $\ulcorner t_i \urcorner$ read off from $\ulcorner \langle t_i \rangle_i \urcorner$.

For the definition of Val_τ, $\tau \in M$, by simultaneous $<$-recursion, we assume that M is a finite set of types and $<$ is a well-ordering of $\mathbf{N}$.

Before we can give a precise definition of the Val_τ, we have to introduce codes of assignments. Let $n + 1 := \max_{\tau \in M} L\tau$. Since we are interested ultimately in closed terms only and have to consider free variables only for their inductive construction, it is sufficient to restrict ourselves to terms with free variables of type level $\leq n$. So let an assignment of functionals $a_1, \ldots, a_m$ to variables $x_1, \ldots, x_m$ be given, both of types $\tau_1, \ldots, \tau_m$, respectively, with levels $L\tau_i \leq n$. First, transform all the a_i to a common standard type n of level n; for this, we use transformation functionals[8] $\mathrm{Tr}^n_{\tau_i}$ with inverses $\mathrm{Tr}^{\tau_i}_n$, such that $\mathrm{Tr}^{\tau_i}_n(\mathrm{Tr}^n_{\tau_i} a_i) = a_i$. With these $\mathrm{Tr}^n_{\tau_i} a_i$, build a tupel $c^n = \langle \ldots, \mathrm{Tr}^n_{\tau_i} a_i, \ldots, \mathbf{0}^n, \ldots \rangle$ of length $\max(\ulcorner x_1 \urcorner, \ldots, \ulcorner x_m \urcorner)$ that has on the $\ulcorner x_i \urcorner$th place just $\mathrm{Tr}^n_{\tau_i} a_i$ (for $i = 1, \ldots, m$) and on all other places $\mathbf{0}^n$. This tupel $c^n =: \ulcorner \mathfrak{x} \to \mathfrak{a} \urcorner$ is the code of the given assignment. From c, one can read off the functional a_i assigned to x_i by $a_i = \mathrm{Tr}^{\tau_i}_n(c)_{\ulcorner x_i \urcorner}$. To an extension (or change) of the given assignment by a requirement that the functional a should be assigned to the variable x there corresponds a

[8] Such transformation functionals can easily be defined explicitly; cf. Gandy [2].

change of the code c to $\mathrm{Ext}(c, \ulcorner x \urcorner, \mathrm{Tr}^n_\tau a)$ with a functional $\mathrm{Ext} = \mathrm{Ext}_n$ that can easily be defined from pr rec functions.

DEFINITION. $\mathrm{Val}_\tau = \mathrm{Val}^{<,M}_\tau$, $\tau \in M$ is defined by:

(i) if $(u)_0 = 1$: $\mathrm{Val}_\tau u^0 c^n = \mathrm{Tr}^\tau_n(c)_u$,

(ii) if $(u)_0 = 2$, $\tau = 0$: $\mathrm{Val}_\tau uc = (u)_3$,

(iii) if $(u)_0 = 3$, $\mathrm{Type}((u)_3) = \ulcorner \sigma \to \tau \urcorner$ with $\sigma \to \tau$, $\sigma \in M$:

$$\mathrm{Val}_\tau uc = (\mathrm{Val}_{\sigma \to \tau}(u)_3 c)(\mathrm{Val}_\sigma (u)_4 c),$$

(iv) if $(u)_0 = 4$, τ of the form $\rho \to \sigma$ with $\rho, \sigma \in M$:

$$\mathrm{Val}_\tau uc = \lambda a^\rho . \mathrm{Val}_\sigma (u)_4(\mathrm{Ext}(c, (u)_3, \mathrm{Tr}^n_\rho a)),$$

(v) if $(u_0) = 5$, $\tau = (0 \to 0)$:

$$\mathrm{Val}_\tau uc = \lambda a^0 . \mathrm{Val}_0([(u)_5](a))c,$$

(vi) otherwise: $\mathrm{Val}_\tau uc = \mathbf{0}^\tau$.

Here all the occurrences of Val in the right hand side of the defining equations are to be replaced by $[\mathrm{Val}]_{|\cdot|<|u|}$, i.e., by a functional with the value $\mathrm{Val}(v)$ at the argument v if $|v| < |u|$, and $\mathbf{0}$ otherwise.

That this definition is reducible to a $<$-recursion can be seen as follows. First it can be reduced to a simultaneous $<$-recursion

$$(*) \qquad F_i v = G_i [F_1]_{<v} \dots [F_k]_{<v} v, \qquad i = 1, \dots, k,$$

by setting $\mathrm{Val}_\tau uc = \mathrm{Val}'_\tau |u| uc$ and defining the $\mathrm{Val}'_\tau vuc$ in the obvious way by simultaneous $<$-recursion, for instance in case (iii) by

$$\mathrm{Val}'_\tau vuc = ([\mathrm{Val}'_{\sigma \to \tau}]_{<v} |(u)_3| (u)_3 c)([\mathrm{Val}'_\sigma]_{<v} |(u)_4| (u)_4 c).$$

Then $(*)$ can be reduced as usual to an ordinary $<$-recursion by[9]

$$F(v) = \langle \mathrm{Tr}^{n+1}_{\tau_i}(F_i v) \rangle_{i=1,\dots,k}.$$

Hence, valuation functionals $\mathrm{Val}_\tau = \mathrm{Val}^{<,M}_\tau$, $\tau \in M$, with the properties (i)–(vi) listed above can be defined by explicit definitions and a $<$-recursion, where all the auxiliary functionals are defined explicitly from pr rec functions.

[9] Even without assuming extensionality a reduction of simultaneous recursion to ordinary recursion is possible, as was shown by Diller and Schütte [1]. However, they use auxiliary functionals of higher type levels and defined by recursion.

3.5. From the definitions of the Val_τ it is clear that $\mathrm{Val}_\tau\ulcorner t^\tau\urcorner\ulcorner \mathfrak{x} \to \mathfrak{a}\urcorner$ always represents the same set theoretic functional as t^τ under the assignment $\mathfrak{x} \to \mathfrak{a}$, namely $\mathrm{Val}^{\mathfrak{a}}_{\mathfrak{x}} t^\tau$.

3.6. Let $<$ be a well-ordering of $\mathbf{N}$ with least element 0. From $<$ one obtains a well-ordering $<'$ with order type $|<'| = 2^{|<|}$ as follows (after Tait[21]): Consider a (easy to define) bijective correspondence

$$a \equiv \langle a_0, \ldots, a_{|a|-1}\rangle$$

between numbers and finite sequences of numbers with

$$a_0 > a_1 > \cdots > a_{|a|-1}.$$

Assume that $|a| = 0$ iff $a = 0$. Let $2^{\alpha_0} + \cdots + 2^{\alpha_n}$ be an ordinal $<2^\alpha$ with $\alpha > \alpha_0 > \cdots > \alpha_n$ and $a_1, \ldots, a_n$ be the numbers corresponding in $<$ to $\alpha_1, \ldots, \alpha_n$. Then we let $\langle a_0, \ldots, a_n\rangle$ be the number corresponding in $<'$ to the ordinal $2^{\alpha_0} + \cdots + 2^{\alpha_n}$. Hence we have

$$a <' b \leftrightarrow \exists k_{k<|a|,|b|}(\forall i_{i<k} a_i = b_i \wedge a_k < b_k) \\ \vee (|a| < |b| \wedge \forall i_{i<|a|} a_i = b_i).$$

If $<$ is pr rec, then clearly so is $<'$.

3.7. Now we show that to the operations on terms given in Section 2–e.g. substitution $t_x[s]$ or reduction $t \dashv t'$ of the rank of t by 1–there correspond pr rec functions between term numbers. So in particular, for the term t^*_F constructed in Section 2.12 from t_F by a sequence of reductions there exists a term number $\ulcorner t^*_F\urcorner$.

Let $<$ be a pr rec well-ordering of $\mathbf{N}$ with least element 0 and $<'$ be the well-ordering of order type $|<'| = 2^{|<|}$ constructed in Section 3.6. The functions corresponding in $<$ to the ordinal functions max, + are denoted by omax, $\oplus$; we assume that they are pr rec. Let M be a set of types.

First we construct a pr rec function $\mathrm{Red} = \mathrm{Red}^<$ such that for any $\ulcorner t\urcorner \in \mathrm{Num}^{<,M}$ with $R\ulcorner t\urcorner \leq k+1$ the following holds.

(1) $\mathrm{Red}(\ulcorner t\urcorner, k) =: \mathrm{Red}_k(\ulcorner t\urcorner) \in \mathrm{Num}^{<',M}$ is a number of the reduced term t' constructed in Section 2.10 from t and k,

(2) $o_{<'}|\mathrm{Red}_k\ulcorner t\urcorner| \leq 2^{o_<|\ulcorner t\urcorner|}$,

(3) $R(\mathrm{Red}_k\ulcorner t\urcorner) \leq k$,

(4) $\mathrm{FV}(\mathrm{Red}_k\ulcorner t\urcorner) \subseteq^* \mathrm{FV}\ulcorner t\urcorner$.

For this we assume that a pr rec function $\mathrm{Sub} = \mathrm{Sub}^<$ is already constructed, such that for any $\ulcorner t \urcorner$, $\ulcorner x \urcorner$, $\ulcorner s \urcorner \in \mathrm{Num}^{<,M}$ we have

(1) $\mathrm{Sub}(\ulcorner t \urcorner, \ulcorner x \urcorner, \ulcorner s \urcorner) \in \mathrm{Num}^{<,M}$ is a number of the term $t_x[s]$,

(2) $|\mathrm{Sub}(\ulcorner t \urcorner, \ulcorner x \urcorner, \ulcorner s \urcorner)| \preccurlyeq |\ulcorner s \urcorner| \oplus |\ulcorner t \urcorner|$,

(3) $R(\mathrm{Sub}(\ulcorner t \urcorner, \ulcorner x \urcorner, \ulcorner s \urcorner)) \leq \max(R\ulcorner t \urcorner, R\ulcorner s \urcorner, L\ulcorner s \urcorner)$,

(4) $\mathrm{FV}(\mathrm{Sub}(\ulcorner t \urcorner, \ulcorner x \urcorner, \ulcorner s \urcorner)) \subseteq^{\#} (\mathrm{FV}\ulcorner t \urcorner -^{\#} \{\ulcorner x \urcorner\}^{\#}) \cup^{\#} \mathrm{FV}\ulcorner s \urcorner$.

By the recursion theorem for $\mathfrak{P}$ there exists a pr rec function Red_k with index e, such that for all u the following holds[10]. If $(u)_0 = 5$:

$$\mathrm{Red}_k(u) = \langle 5, \mathrm{Type}(u), \ulcorner 2^{0_{\cdot}(|u|)} \urcorner, k, \mathrm{FV}(u), e' \rangle,$$

where $e' = e'(e, u)$ is a pr rec index of $\mathrm{Red}_k([(u)_5](n))$ as a function of n; e' as a function of e and u is pr rec. If $(u)_0 = 4$: $\mathrm{Red}_k(u) = \langle 4, \mathrm{Type}(u), |v| \oplus \ulcorner 1 \urcorner, (u)_3, v \rangle$ with $v = \mathrm{Red}_k((u)_4)$. If $(u)_0 = 3$: Let $\mathrm{Red}_k((u)_3) = v$ and $\mathrm{Red}_k((u)_4) = w$. If the type level $L(v) = k+1$ and $(v)_0 = 4$, define $\mathrm{Red}_k(u) = \mathrm{Sub}^{<'}((v)_4, (v)_3, w)$. If $L(v) \neq k+1$ or $(v)_0 \neq 4$, define $\mathrm{Red}_k(u) = \langle 3, \mathrm{Type}(u), \mathrm{omax}(|v|, |w|) \oplus \ulcorner 1 \urcorner, v, w \rangle$. Otherwise: $\mathrm{Red}_k(u) = u$. Comparing this definition with Section 2.10 one can see easily that Red_k has the required properties.

Next we construct a function $\mathrm{Sub} = \mathrm{Sub}^{<}$ with the properties listed above. For this we assume this time that a pr rec function C (C refers to change of bound variables) is already constructed such that for any $\ulcorner t \urcorner$, $\ulcorner x \urcorner \in \mathrm{Num}^{<,M}$ we have (1) $C(\ulcorner t \urcorner, \ulcorner x \urcorner) \in \mathrm{Num}^{<,M}$ is a number of a term t_1 obtained from t by a change of bound variables such that t_1 does not contain a bound occurrence of x, (2) $|C(\ulcorner t \urcorner, \ulcorner x \urcorner)| = |\ulcorner t \urcorner|$ and (3) $\mathrm{FV}(C(\ulcorner t \urcorner, \ulcorner x \urcorner)) = \mathrm{FV}\ulcorner t \urcorner$.

We define $\mathrm{Sub}(u, v, w)$ in the form $\mathrm{Sub}_1(C^*(u, \mathrm{FV}(w)), v, w)$, where C^* is a simple variant of C working with a finite set of variables instead of a single variable (hence $C^*(u, \{\ulcorner x_1 \urcorner, \ldots, \ulcorner x_n \urcorner\}^{\#}) = C(\ldots C(u, \ulcorner x_1 \urcorner), \ldots, \ulcorner x_n \urcorner))$. A pr rec function Sub_1 with index e is constructed again by the recursion theorem for $\mathfrak{P}$, as a solution of the following equations. If $(u)_0 = 5$:

$$\mathrm{Sub}_1(u, v, w) = \langle 5, \mathrm{Type}(u), |w| \oplus |u|, \max(R(u), R(w). L(w)), (\mathrm{FV}(u) -^{\#} \{v\}^{\#}) \cup^{\#} \mathrm{FV}(w), e' \rangle,$$

where $e' = e'(e, u, v, w)$ is a pr rec index of $\mathrm{Sub}_1([(u)_5](n), v, w)$ as a function of n. If $(u)_0 = 4$: $\mathrm{Sub}_1(u, v, w) = \langle 4, \mathrm{Type}(u), |u_4| \oplus \ulcorner 1 \urcorner, (u)_3, u_4 \rangle$ with $u_4 = \mathrm{Sub}_1((u)_4, v, w)$. If $(u)_0 = 3$: $\mathrm{Sub}_1(u, v, w) = \langle 3, \mathrm{Type}(u),$

[10] Cf. the definition of $+'_0$ in Kleene[8, p. 75].

omax($|u_3|$, $|u_4|)\oplus\ulcorner 1\urcorner$, u_3, $u_4\rangle$ with $u_i = \mathrm{Sub}_1((u)_i, v, w)$ for $i = 3, 4$. If $(u)_0 = 1$, $v = u$: $\mathrm{Sub}_1(u, v, w) = w$. Otherwise: $\mathrm{Sub}_1(u, v, w) = u$.–Again one can see easily that Sub_1 has the required properties.

For the construction of C we make use of a pr rec function Repl (for replacement) with the following properties: For any $\ulcorner t\urcorner, \ulcorner x^\sigma\urcorner, \ulcorner y^\sigma\urcorner \in \mathrm{Num}^{<, M}$,

(1) $\mathrm{Repl}(\ulcorner t\urcorner, \ulcorner x\urcorner, \ulcorner y\urcorner) \in \mathrm{Num}^{<, M}$ is a number of a term obtained from t by replacing all (free and bound) occurrences of x by y,

(2) $|\mathrm{Repl}(\ulcorner t\urcorner, \ulcorner x\urcorner, \ulcorner y\urcorner)| = |\ulcorner t\urcorner|$,

(3) $\mathrm{FV}(\mathrm{Repl}(\ulcorner t\urcorner, \ulcorner x\urcorner, \ulcorner y\urcorner)) = (\mathrm{FV}\ulcorner t\urcorner -^{\#} \{\ulcorner x\urcorner\}^{\#}) \cup^{\#} \{\ulcorner y\urcorner\}^{\#}$.

The definition of C from Repl and the definition of Repl can be carried out as before by the recursion theorem for $\mathfrak{P}$; we omit the details.

3.8. Let $<$ be a pr rec well-ordering of **N** with least element 0, such that to the ordinal functions $+, \cdot$ there correspond pr rec functions. For simplicity (i.e., for to be able to reduce primitive recursion to $<$-recursion) we again assume (1), (2) of Section 1.6.

THEOREM 1. *Let a functional F of type level $n+1$ be defined by explicit definitions and $<$-recursions. Assume that the type level of any auxiliary functional introduced by recursion is $\leq n+m+1 (m \geq 1)$. Then one can find a new definition of F, containing auxiliary functionals of type level $\leq n+1$ only, but using instead of the $<$-recursion a $<^*$-recursion with a well-ordering $<^*$ constructed canonically from $<$ and of an order type $|<^*| < 2_m(|<|\omega)$ (where $2_0(\xi) := \xi$, $2_{i+1}(\xi) := 2^{2_i(\xi)}$).*

PROOF. By M_k we denote the set of all types used in the definition of F and with levels $\leq k$. In Section 3.3 we have defined a number $\ulcorner t_F\urcorner \in \mathrm{Num}^{<_0, M_{n+m+1}}$ of an infinite term t_F representing F. $<_0$ was a well-ordering constructed canonically from $<$ and of order type $|<_0| = |<|\omega$. For fixed F, one clearly can use instead of $<_0$ a well-ordering $<_l$ constructed canonically from $<$ and of order type $|<_l| = |<|l$ (with suitable l). Hence $\ulcorner t_F\urcorner \in \mathrm{Num}^{<_l, M_{n+m+1}}$. Furthermore, we had $R\ulcorner t_F\urcorner \leq n+m+1$ and $\mathrm{FV}\ulcorner t_F\urcorner = \emptyset^{\#}$. By applying m times the reduction function constructed in Section 3.7, we now obtain a number

$$\begin{aligned} \mathrm{Red}_{n+1}(\mathrm{Red}_{n+2} \ldots (\mathrm{Red}_{n+m}\ulcorner t_F\urcorner) ..)) &=: \mathrm{Red}^*\ulcorner t_F\urcorner \\ &=: \ulcorner t_F^*\urcorner \\ &\in \mathrm{Num}^{<^*, M_{n+m+1}} \end{aligned}$$

of the t_F^* constructed in Section 2.12. Here $<^*$ is the well-ordering obtained from $<_1$ by applying m times the $'$-operation of Section 3.6 and hence of an order type $|<^*| < 2_m(|<|\omega)$. From the properties of the reduction function we conclude $R\ulcorner t_F^* \urcorner \leq n+1$ and $\mathrm{FV}\ulcorner t_F^* \urcorner = \emptyset^*$. So we have $\ulcorner t_F^* \urcorner \in \mathrm{Num}^{<^*, M_{n+1}}$ too. As a new definition of F we choose

$$F^\tau = \mathrm{Val}_\tau^{<^*, M_{n+1}} \ulcorner t_F^* \urcorner;$$

this is possible since (cf. Section 3.5) $\mathrm{Val}^{<^*, M_{n+1}} \ulcorner t_F^* \urcorner$ denotes the same set theoretic functional as t_F^*, namely F. The required properties of this new definition follow immediately from the definition of the valuation functionals in Section 3.4.

REMARK. It is possible to have all the auxiliary functionals in the new definition of F (i.e., not only those introduced by recursion) of type level $\leq n+1$, if one modifies a little the notion of $<$-recursion, namely to $F(x) = A[[F]_{<x}, x]$ with a finite term A. The proof is obtained immediately from the normal form theorem for finite terms. A modification of the recursion schema is needed, since in $F(x) = G([F]_{<x}, x)$ G has necessarily a higher type level than F.

4. Formalization in extensions of Gödel's T

We want to extend Theorem 1 to more general type structures instead of the set theoretic functionals. So we consider arbitrary models of the theory T of pr rec functionals of Gödel [5] in the extensional version of Spector [18], extended by adding $<$-recursion and $<$-induction. We show that Theorem 1 is true in this general context too, and that the proof can be carried out quantifier-free, more precisely within the considered extension of T.

4.1. So let T be as above and for a well-ordering $<$ of $\mathbf{N}$ let $T_<$ be the extension of T by $<$-recursion $F(x) = G([F]_{<x}, x)$ and $<$-induction

$$\frac{F_i(x) = G([F_i]_{<x}, x) \qquad i = 1, 2}{F_1(x) = F_2(x).}$$

Here $[F]_{<x}$ stands for $\lambda y_{y<x} F y$, i.e., for $\lambda y D(Fy, \mathbf{0}, c_<(y, x))$ with $D(x^\tau, y^\tau, z^0) := x$ if $z = 0$, and $:= y$ otherwise. From the chosen form of

$<$-induction one obtains easily other forms, e.g. the following

$$\frac{\begin{array}{c} P(0, y) \\ \bigwedge_{i=1}^{n} P([g_i xy]_{<x}, hxy) \to P(x, y) \end{array}}{P(x, y)}.$$

Here $[g_i xy]_{<x}$ stands for $D(g_i xy, 0, c_<(g_i xy, x))$, and it is assumed that $0 < x$ for all $x \neq 0$.

On $<$ we again make the assumptions (1), (2) of Section 1.6. The reduction of primitive recursion to $<$-recursion given there can be carried out in $T_<$ too. Hence we do not need an extra schema of primitive recursion in $T_<$.

By $T^n_<$ we denote the subtheory of $T_<$ obtained by restricting all constants introduced by recursion to those of type level $\leq n+1$. Let $T^n_< + PL[HA^n_<]$ be the extension of $T^n_<$ by many sorted intuitionistic predicate logic without [with] the (ordinary) induction rule for the extended language. $T^n_< + PL$ is a conservative extension of $T^n_<$, as one can prove easily by means of the Gödel interpretation (see e.g. Spector[18]). Note that in this proof auxiliary functionals of type level $> n+1$ may occur, but all of them are explicitly defined (this is not the case for $HA^n_<$).

4.2. Under the same assumptions on $<$ as in Section 3.8 we have

THEOREM 2. *For any constant F in $T^{n+m}_<$ $(m \geq 1)$ of type level $n+1$ one can find a constant F' in $T^n_{<^*}$ with a well-ordering $<^*$ of order type $|<^*| < 2_m(|<|\omega)$ constructed canonically from $<$, such that $Fx_1 \dots x_k = F'x_1 \dots x_k$ is provable in $T^{n+m}_{<^*}$.*

PROOF. Let $<_1$, $<^*$, the M_k and a $\ulcorner t_F \urcorner \in \mathrm{Num}^{<_1, M_{n+m+1}}$ with $R\ulcorner t_F \urcorner \leq n+m+1$, $\mathrm{FV}\ulcorner t_F \urcorner = \emptyset^{\#}$ be chosen as in Section 3.8. We will show in Section 4.3,

$$T^{n+m}_{<_1} \vdash F = \mathrm{Val}^{<_1, M_{n+m+1}} \ulcorner t_F \urcorner \mathbf{0}$$

and in Section 4.4 the following *Reduction Lemma*: Let M be a finite set of types with levels $\leq i+1$, and $\tau \in M$. Then one can derive in $T^i_< + PL$

$$\begin{aligned} & u \in \mathrm{Num}^{<, M} \wedge R(u) \leq k+1 \wedge \mathrm{Type}(u) = \ulcorner \tau \urcorner \\ & \to \mathrm{Red}_k(u) \in \mathrm{Num}^{<', M} \\ & \quad \wedge |\mathrm{Red}_k(u)| <' \ulcorner 2^{o_<(|u|)} \urcorner \end{aligned}$$

$$\text{(Red)} \qquad \begin{aligned} &\wedge\ R(\mathrm{Red}_k(u)) \leq k \\ &\wedge\ \mathrm{FV}(\mathrm{Red}_k(u)) \subseteq^{\#} \mathrm{FV}(u) \\ &\wedge\ \mathrm{Type}(\mathrm{Red}_k(u)) = \ulcorner \tau \urcorner \\ &\wedge\ \mathrm{Val}_\tau^{<, M}\, u = \mathrm{Val}_\tau^{<', M}(\mathrm{Red}_k\, u). \end{aligned}$$

Furthermore we show in Section 4.4, that under the same assumptions on M one can derive in $\mathrm{T}^i_< + \mathrm{PL}$,

$$(*) \qquad \begin{aligned} &u \in \mathrm{Num}^{<, M} \wedge R(u) \leq k \wedge \forall v \in^{\#} \mathrm{FV}(u)\colon \mathrm{Type}(v) \in M_k \\ &\wedge\ \mathrm{Type}(u) = \ulcorner \tau \urcorner \to \mathrm{Val}_\tau^{<, M}\, u = \mathrm{Val}_\tau^{<, M_k}\, u \end{aligned}$$

with an arbitrary set $M_k \subseteq M$ of types of level $\leq k$.–From this, Theorem 2 follows, since one has in $\mathrm{T}^{n+m}_{<^*}$

$$\begin{aligned} F &= \mathrm{Val}^{<_1, M_{n+m-1}} \ulcorner t_F \urcorner \mathbf{0} \\ &= \mathrm{Val}^{<^*, M_{n+1}}(\mathrm{Red}_{n+1}(\mathrm{Red}_{n+2} \ldots (\mathrm{Red}_{n+m} \ulcorner t_F \urcorner) \ldots))\mathbf{0} \end{aligned}$$

and hence a representation of F by a constant of $\mathrm{T}^n_{<^*}$.

4.3. PROOF OF $\mathrm{T}^{n+m}_{<_1} \vdash F = \mathrm{Val}^{<_1, M_{n+m+1}} \ulcorner t_F \urcorner \mathbf{0}$. We make use of some simple properties of the valuation functionals which we list first. A denotes a finite term (i.e., built up from variables and numerals by application and abstraction), and $\mathrm{Val}^{\mathfrak{a}}_{\mathfrak{x}}\, u$ stands for

$$\mathrm{Val}(u, \mathrm{Ext}(\ldots \mathrm{Ext}(\mathbf{0}, \ulcorner x_1 \urcorner, \mathrm{Tr}^{n+m}_{\tau_1}\, a_1) \ldots, \ulcorner x_p \urcorner, \mathrm{Tr}^{n+m}_{\tau_p}\, a_p))$$

(the parameters $<_l$, M_{n+m+1} of Val are omitted).

(1) $\mathrm{Val}^{\mathfrak{a} a b \mathfrak{b}}_{\mathfrak{x} x y \mathfrak{y}} \ulcorner A \urcorner = \mathrm{Val}^{\mathfrak{a} b a \mathfrak{b}}_{\mathfrak{x} y x \mathfrak{y}} \ulcorner A \urcorner$,

(2) $\mathrm{Val}^{\mathfrak{a} a \mathfrak{b} d \mathfrak{c}}_{\mathfrak{x} x \mathfrak{y} x \mathfrak{z}} \ulcorner A \urcorner = \mathrm{Val}^{\mathfrak{a} \mathfrak{b} d \mathfrak{c}}_{\mathfrak{x} \mathfrak{y} x \mathfrak{z}} \ulcorner A \urcorner$,

(3) $\mathrm{Val}^{\mathfrak{a} a}_{\mathfrak{x} x} \ulcorner A \urcorner = \mathrm{Val}^{\mathfrak{a}}_{\mathfrak{x}} \ulcorner A \urcorner$ if x is not free in A,

(4) $\mathrm{Val}^{\mathfrak{a}}_{\mathfrak{x}} \ulcorner A_x[t] \urcorner = \mathrm{Val}^{\mathfrak{a}\, \mathrm{Val}^{\mathfrak{a}}_{\mathfrak{x}} \ulcorner t \urcorner}_{\mathfrak{x}\, x} \ulcorner A \urcorner$,

(5) $\mathrm{Val}^{\mathfrak{a}}_{\mathfrak{x}} \ulcorner A \urcorner = A_{\mathfrak{x}}[\mathfrak{a}]$ if $\mathfrak{x}$ contains all the free variables in A.

(1)—(5) are obtained easily by induction on A.

For the proof of $F = \mathrm{Val} \ulcorner t_F \urcorner$ in $\mathrm{T}^{n+m}_{<_1}$ we restrict ourselves to the case of an F introduced by $<$-recursion, the other cases being easier or trivial. Then F fulfills a recursion equation (cf. Section 3.3)

$$F(x) = A^*[[F]_{<x}, x, G_1, \ldots, G_p].$$

It suffices to show that $\mathrm{Val} \ulcorner t_F \urcorner$ fulfills the same recursion equation

$$\mathrm{Val} \ulcorner t_F \urcorner x = A^*[[\mathrm{Val} \ulcorner t_F \urcorner]_{<x}, x, G_1, \ldots, G_p],$$

for then by $<$-induction we can conclude $F = \mathrm{Val}\ulcorner t_F \urcorner$. One obtains first $\mathrm{Val}\ulcorner t_F \urcorner x = \mathrm{Val}([e](x))$ with an e (as in Section 3.3) such that $[e](x)$ is a term number of $A^*[\langle t_{xi}\rangle_i, \bar{x}, t_{G_1}, \ldots, t_{G_p}]$ built up from term numbers $\ulcorner t_{G_1}\urcorner, \ldots, \ulcorner t_{G_p}\urcorner$ for $G_1, \ldots, G_p$ and $e' = e'(e, x)$ (cf. Section 3.3) for $\langle t_{xi}\rangle_i$. Hence by (4) and (5),

$$\mathrm{Val}\ulcorner t_F \urcorner x = A^*[\mathrm{Val}(e'(e, x)), x, \mathrm{Val}\ulcorner t_{G_1}\urcorner, \ldots, \mathrm{Val}\ulcorner t_{G_p}\urcorner].$$

By hyp. of the induction on F we already have $\mathrm{Val}\ulcorner t_{G_i}\urcorner = G_i$. Hence it suffices to show

$$\mathrm{Val}(e'(e, x))y = [\mathrm{Val}\ulcorner t_F \urcorner]_{<x} y.$$

This is proved immediately by distinguishing the cases $y < x$ and $y \not< x$.

4.4. First we need a representation of $\mathrm{Num}^{<, M}$ in Π^0_1-form, which can be obtained as follows. Infinite terms may be considered as well-founded trees, where at each node there is either no branching at all (i.e., it is a bottommost node) and a variable or a numeral is affixed, or there is a 2-fold branching (this corresponds to application), or a 1-fold branching with a variable affixed (this corresponds to λ-abstraction with this variable), or an ω-fold branching (this corresponds to the formation of sequences). Then any term number $\ulcorner t \urcorner$ can be thought of as obtained inductively by affixing to each node of the tree corresponding to t a term number of the corresponding subterm. Hence the property $u \in \mathrm{Num}^{<, M}$ is equivalent to u having such a well-founded genealogic tree. But the latter fact can be written easily in Π^0_1-form: One has to express that at any node (= sequence number) n the tree is locally correct, i.e., that the term number u_n affixed there (u_n can be defined easily by induction on n) and all its predecessors $u_{n*\langle i\rangle}$, $i = 0, 1, 2, \ldots$, fulfill a relation as given in the definition of term numbers. The well-foundedness is then obtained automatically, since in particular $|u_{n*\langle i\rangle}| < |u_n|$ is required and $<$ is a well-ordering.–The representation of $u \in \mathrm{Num}^{<, M}$ gotten this way has the form $\forall x\, P(x, u)$ with a predicate P pr rec in the enumeration function $\lambda ab\,.\,[a](b)$ of $\mathfrak{P}$.

Now we obtain that the formula (Red) occurring in the Reduction Lemma and the formula (*) can be derived in $\mathrm{HA}^i_<$, under the assumptions formulated in Section 4.2. In both cases the derivation is obtained easily by $<$-induction on $|u|$, if (in the case (Red)) one assumes (cf. Section 3.7)

$$u, v, w \in \mathrm{Num}^{<, M} \wedge \mathrm{Var}(v) \wedge \mathrm{FV}(u) \cap^{\#} \mathrm{FV}(w) = \emptyset^{\#}$$

$$\begin{aligned}
&\wedge \text{Type}(u) = \ulcorner\tau\urcorner \wedge \text{Type}(v) = \text{Type}(w) = \ulcorner\sigma\urcorner \\
&\to \text{Sub}_1(u, v, w) \in \text{Num}^{<,M} \\
&\quad \wedge |\text{Sub}_1(u, v, w)| \preccurlyeq |w| \oplus |u| \\
\text{(Sub)} \qquad &\quad \wedge R(\text{Sub}_1(u, v, w)) \leq \max(Ru, Rw, Lw) \\
&\quad \wedge \text{FV}(\text{Sub}_1(u, v, w)) \subseteq^{\#} (\text{FV}(u) -^{\#} \{v\}^{\#}) \cup^{\#} \text{FV}(w) \\
&\quad \wedge \text{Type}(\text{Sub}_1(u, v, w)) = \ulcorner\tau\urcorner \\
&\quad \wedge \text{Val}_\tau^{<,M}(\text{Sub}_1\, uvw) = \text{Val}_\tau^{<,M}(u, \text{Ext}(c, v, \text{Val}_\sigma^{<,M}\, wc)).
\end{aligned}$$

This too is proved easily by $<$-induction on $|u|$, if a corresponding formula (C) for the function C is available (cf. Section 3.7), and (C) in turn is obtained by $<$-induction on $|u|$ from a corresponding formula (Repl) for the function Repl (cf. Section 3.7), which finally can be proved directly by $<$-induction on $|u|$.

From these particular derivations of (Red) and ($*$) in $\text{HA}^i_<$ we now construct by means of Herbrand's Theorem the required derivations in $\text{T}^i_< + \text{PL}$ (cf. Kreisel [9] and Shepherdson [17]). For this, note first that all the quantified formulas derived by $<$-induction, namely ($*$), (Red), (Sub), (C) and (Repl), have the form

$$\forall x\, P(x, u) \to Q(u, \mathfrak{y})$$

with quantifier-free P, Q. In every case, this is derived by means of intuitionistic predicate logic from

$$\forall \mathfrak{y}\, \forall v_{|v|<|u|}[\forall x\, P(x, v) \to Q(v, \mathfrak{y})],$$

from closures of quantifier-free formulas and from already derived formulas of the form (Sub), (C), (Repl). We want to construct each time a function f, such that $P(f(u, \mathfrak{y}), u) \to Q(u, \mathfrak{y})$ is derivable in $\text{T}^i_<$. For this we can start from a derivation in intuitionistic predicate logic of the formula

$$\begin{aligned}
&\forall \mathfrak{z}\, R(\mathfrak{z}) \wedge \forall \mathfrak{y}, v\, \exists x\, [|v| < |u| \wedge P(x, v) \to Q(v, \mathfrak{y})] \\
&\wedge \forall x\, P(x, u) \to Q(u, \mathfrak{y})
\end{aligned}$$

in the language of $\text{T}^i_< + \text{PL}$, with quantifier-free R. For simplicity, we omit the parameters $\mathfrak{y}$ and assume $\mathfrak{z} \equiv z$. By Herbrand's Theorem there are terms

$$\begin{aligned} s_i &\equiv s_i(u, x_1, \ldots, x_{i-1}) \quad &&\text{for } i = 1, \ldots, n, \\ t_j &\equiv t_j(u, x_1, \ldots, x_n) \quad &&\text{for } j = 1, \ldots, m, \\ r_k &\equiv r_k(u, x_1, \ldots, x_n) \quad &&\text{for } k = 1, \ldots, l, \end{aligned}$$

of type 0, such that

$$\bigwedge_k R(r_k) \wedge \bigwedge_i [|s_i| < |u| \wedge P(x_i, s_i) \to Q(s_i)] \\ \wedge \bigwedge_j P(t_j, u) \to Q(u)$$

is derivable in propositional logic. Hence in $T^i_<$ we can derive

(1) $$\bigwedge_i [|s_i| < |u| \wedge P(x_i, s_i) \to Q(s_i)] \to \bigvee_j [P(t_j, u) \to Q(u)].$$

Now we define a function f by the following $<$-recursion on $|u|$

$$f(u) = \min\{t_j(u, x_1, \ldots, x_n) \colon P(t_j(u, x_1, \ldots, x_n), u) \to Q(u)\}$$

with

$$x_i = \begin{cases} f(s_i(u, x_1, \ldots, x_{i-1})) & \text{if } |s_i(u, x_1, \ldots, x_{i-1})| < |u|, \\ 0 & \text{otherwise.} \end{cases}$$

Then from (1) one obtains in $T^i_<$

$$\bigwedge_i [|s_i| < |u| \wedge P(f(s_i), s_i) \to Q(s_i)] \to [P(f(u), u) \to Q(u)],$$

i.e., the premiss of a $<$-induction on $|u|$, which can be brought easily in the form mentioned in Section 4.1. Hence $P(f(u), u) \to Q(u)$ is derivable in $T^i_<$.

References

[1] J. Diller and K. Schütte, Simultane Rekursionen in der Theorie der Funktionale endlicher Typen, *Archiv für mathematischen Logik* **14** (1971) 69–74.

[2] R. O. Gandy, Computable functionals of finite type I, in: J. N. Crossley, ed., *Sets, Models and Recursion Theory* (North-Holland, Amsterdam, 1967) pp. 202–242.

[3] G. Gentzen, Die Widerspruchsfreiheit der reinen Zahlentheorie, *Mathematische Annalen* **112** (1936) 493–565.

[4] G. Gentzen, Neue Fassung des Widerspruchsfreiheitsbeweises für die reine Zahlentheorie, in: *Forschungen zur Logik und zur Grundlegung der exakten Wissenschaften*, Neue Folge 4 (Hirzel, Leipzig, 1938) pp. 19–44.

[5] K. Gödel, Über eine bisher noch nicht benützte Erweiterung des finiten Standpunkts, *Dialectica* **12** (1958) 280–287.
[6] O. Hilbert and P. Bernays, *Grundlagen der Mathematik*, Vols. I and II (Springer, Berlin, 1934 and 1939).
[7] W. A. Howard, Assignment of ordinals to terms for primitive recursive functionals of finite type, in: J. Myhill, A. Kino, R. E. Vesley, eds., *Intuitionism and Proof Theory* (North-Holland, Amsterdam, 1970) pp. 443–458.
[8] S. C. Kleene, Extension of an effectively generated class of functions by enumeration, *Colloquium Mathematicum* **7** (1958) 67–78.
[9] G. Kreisel, Mathematical significance of consistency proofs, *Journal of Symbolic Logic* **23** (1958) 155–182.
[10] G. Kreisel, Proof by transfinite induction and definition by transfinite induction in quantifier-free systems (abstract), *Journal of Symbolic Logic* **24** (1959) 322.
[11] G. Kreisel, Inessential extensions of Heyting's arithmetic by means of functionals of finite type (abstract), *Journal of Symbolic Logic* **24** (1959) 284.
[12] G. Kreisel, Interpretation of analysis by means of constructive functionals of finite types, in: A. Heyting, ed., *Constructivity in Mathematics*, Proc. Colloquium Amsterdam, 1957 (Amsterdam, 1959) pp. 101–128.
[13] G. Kreisel, Mathematical logic, in: T. L. Saaty, ed., *Lectures on Modern Mathematics*, Vol. III (Wiley, New York, 1965) pp. 95–195.
[14] J. Myhill, A stumblingblock in constructive mathematics (abstract), *Journal of Symbolic Logic* **18** (1953) 190.
[15] C. Parsons, Proof theoretic analysis of restricted induction schemata (abstract), *Journal of Symbolic Logic* **36** (1971) 361.
[16] Routledge, Ordinal recursion, *Proceedings of the Cambridge Philosoph. Society* **49** (1953) 175–182.
[17] J. C. Shepherdson, Non-standard models for fragments of number theory, in: J. W. Addison, L. Henkin, A. Tarski, eds., *The Theory of Models* (North-Holland, Amsterdam, 1965) pp. 342–358.
[18] C. Spector, Provably recursive functionals of analysis: a consistency proof of analysis by an extension of principles formulated in current intuitionistic mathematics, in: J. Dekker, ed., *Recursive Function Theory*, Proceedings of the 5th Symposium in Pure Mathematics, New York, 1961 (Am. Math., Soc. Providence, R. I., 1962) pp. 1–27.
[19] W. W. Tait, Infinitely long terms of transfinite type, in: J. N. Crossley and M. A. E. Dummet, eds., *Formal Systems and Recursive Functions* (North-Holland, Amsterdam, 1965) pp. 176–185.
[20] W. W. Tait, Functionals defined by transfinite recursion, *Journal of Symbolic Logic* **31** (1965) 155–174.
[21] W. W. Tait, Constructive reasoning, in: B. van Rootselaar and J. F. Staal, eds., *Logic, Methodology and the Philosophy of Science* III (North-Holland, Amsterdam, 1967) pp. 185–199.