AUT 90

Ju 10

EINDHOVEN UNIVERSITY OF TECHNOLOGY

Department of Mathematics
and Computing Science

Memorandum 1985-02

THE LANGUAGE THEORY OF Λ_∞,
A TYPED λ-CALCULUS WHERE TERMS ARE TYPES

by

L.S. van Benthem Jutting

Eindhoven University of Technology,
Department of Mathematics and
Computing Science,
P.O. Box 513,
5600 MB Eindhoven,
The Netherlands.

THE LANGUAGE THEORY OF Λ_∞,

A TYPED λ-CALCULUS WHERE TERMS ARE TYPES

by

L.S. van Benthem Jutting

1. Introduction

In the present paper we present the theory of a system op typed λ-calculus Λ_∞, which is essentially the system introduced by Nederpelt in [6]. Its characteristic feature is that any term of the system can serve as a type. The main difference between the two systems is that our system only allows for β-reduction, while Nederpelt's system has η-reduction as well.

The importance of Λ_∞ lies in the fact that it may be considered as basic to the AUTOMATH languages. Therefore its theory can also be seen as basic to the theory of AUTOMATH [2,3].

In our notation we will follow the habits of AUTOMATH, that is:

for terms u and v, types α and variables x we will denote

$\lambda x^\alpha u$ by $[x:\alpha]u$

and

(uv) by $\langle v\rangle u$.

The system consisting of such terms will be called Δ. The system Λ_∞ is the subset of Δ to which a term $\langle u\rangle v$ belongs only if v is a function, and if the domain of v and the type of u have a common (β-)reduct.

Our main theorems will be:

1. Church-Rosser for Δ. This will be proved along the lines of well-known proofs by Tait and Martin-Löf [5].
2. Strong normalization for a subsystem of "normable terms" in Δ.

Our proof will be along the lines of proofs by Gandy [4] and de Vrijer [7] for strong mormaization in simple typed λ-calculus.

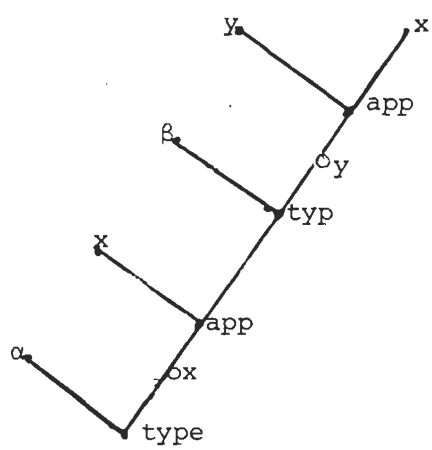
3. Closure of Λ_∞ under (β-)reduction. For this we have a new direct proof, though the theorem has been proved previously by van Daalen [3] .

Moreover, we prove that the terms of Λ_∞ are "normable" in the sense intended above; therefore those terms strongly normalize. This, together with correctness of types, implies that Λ_∞ is decidable.

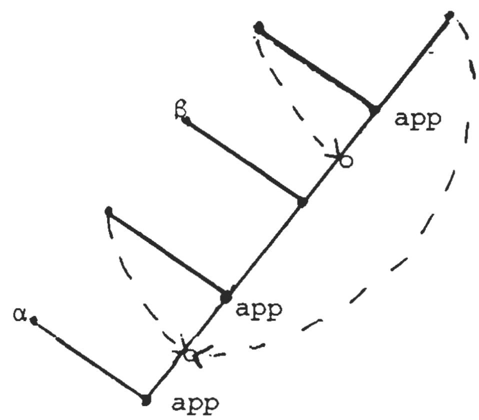
In our presentation we will use "nameless variables" as suggested by de Bruijn [1]. That is, our variables will not be "letters from an alphabet" but "references to a binding λ", or rather, because of our notational habits, "references to a binding square brackets pair". In order to grasp the use of nameless variables one should note that terms can be interpreted as trees. Consider e.g. the term:

$$[x:\alpha] < x > [y:\beta] < y > x \ .$$

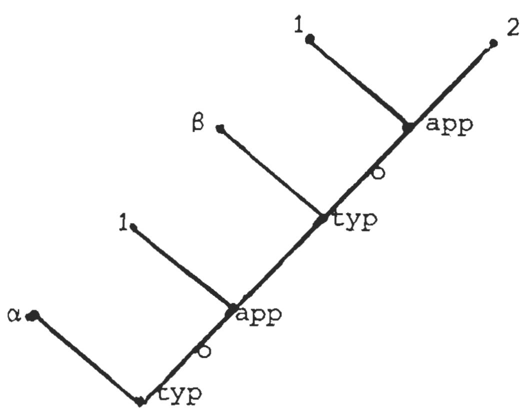
The corresponding tree is



In this tree the bindings may be indicated by arrows, omitting the names of the variables :



and here, again the arrows may be replaced by numbers, indicating the depth of the binding node to which the arrow points as seen from the node where the arrow starts (only binding nodes, indicated by "o" , are counted!):



This last tree can again be represented in a linear form:

$[\alpha] < 1 > [\beta] < 1 > 2$.

Note that the same variable x in the first term (or tree) is represented in the "nameless" term (or tree) once by 1 and once by 2, whereas the same reference 1 in the "nameless" representation once denotes x and once y. Both the name carrying and the nameless linear representation can be considered as formalizations of the underlying intuitive notion of "tree with arrows".

The presentation with nameless variables makes the notion of α-conversion superfluous (and even meaningless). Thereby the definition of operations where "clash of variables" might arise (e.g. substitution) becomes more definite, and the proofs more formal. The drawbacks of this presentation might be a loss of "readability" of the formulas, and the need of a number of technical lemmas for updating references involved in certain formula manipulations.

In our presentation frequent use will be made of inductive definitions (e.g. the definition of term, of substitution, of reduction and of Λ_∞). Subsequently proofs are given with induction with repect to these definitions. This should always be understood in the sense of "induction with respect to the number of applications of a clause in the definition", or, in other words, "induction with respect to the derivation tree". This concept is not formalized here.

2. Preliminaries and notations

In our theory we will use some notions of intuitive set theory. $\mathbb{N}$ will denote the set of natural numbers $\{0,1,2,3,\ldots.\}$, $\mathbb{N}^+$ the set of positive natural numbers $\{1,2,3,\ldots\}$, and $\mathbb{N}^\infty = \mathbb{N} \cup \{\infty\}$ the set N extended with infinity. The predecessor function is extended to $\mathbb{N}^\infty$ by defining $\infty - 1 := \infty$.

For $n \in \mathbb{N}$ we define $\mathbb{N}_n := \{k \in \mathbb{N}^+ \mid k \leq n\}$, so $\mathbb{N}_0 = \emptyset$, the empty set.
Let A and B be sets, Then $A \times B$ denotes the cartesian product of A and B, that is the set of pairs $\lceil a,b \rceil$ where $a \in A$ and $b \in B$; and $A \rightarrow B$ denotes the set of functions with domain A and values in B. If $f \in A \rightarrow B$ and $a \in A$ then $\langle a \rangle f$ will denote the value of f at a; and if for $a \in A$ we have $b(a) \in B$ then $[a \in A]b(a)$ will denote the corresponding function, that is the set $\{\lceil a,b(a) \rceil \in A \times B \mid a \in A\}$.

As a consequence of our notation for the values of a function our notation for the composition of functions will be a little unusual: if f and g are functions with domains A and B respectively, then

$$f \circ g = [x \in C] \langle\langle x \rangle f \rangle g \text{ ,where } C = \{x \in A \mid \langle x \rangle f \in B\}.$$

So $\langle x \rangle (f \circ g) = \langle\langle x \rangle f \rangle g$ for $x \in C$.

If A is a collection of sets then $\cup A$ denotes the union of A.

If A is any set and $n \in \mathbb{N}$ then $A^{(n)}$ denotes $\mathbb{N}_n \to A$, i.e. the set of finite sequences of elements of A with length n. In particular $A^{(0)} = \{\emptyset\}$ and $\emptyset$ is the empty sequence. A^* will denote $\cup \{A^{(n)} \mid n \in \mathbb{N}\}$, that is the set of all finite sequences of elements of A. If $s \in A^*$ then L(s) is the length of a; and if $s1 \in A^*$ and $s2 \in A^*$ then s1&s2 denotes the concatenation of s1 and s2. In particular, $\emptyset \& s = s$ for $s \in A^*$.

If $a \in A$ we will often confuse a with $\{[1,a]\}$, that is the element of $A^{(1)}$ with value a. In particular, if $a \in A$ and $s \in A^*$, then $a\&s \in A^*$,

$$\langle 1 \rangle (a\&s) = a, \text{ and } \langle n+1 \rangle (a\&s) = \langle n \rangle s \text{ for } n \leq L(s).$$

Where no confusion is expected we will often omit the symbol "&".

For the updating of references we will use the following functions and operations on functions:

For $m \in \mathbb{N}$ $\qquad \varphi_m = [n \in \mathbb{N}^+](n + m)$.

For $m \in \mathbb{N}$ $\qquad \Theta_m = [n \in \mathbb{N}^+] T(m,n)$, where

$$T(m,n) = \begin{cases} n + 1 & \text{if } n \leq m \\ 1 & \text{if } n = m+1 \\ n & \text{if } n > m+1. \end{cases}$$

For $m \in \mathbb{N}$ and $\psi \in \mathbb{N}^+ \to \mathbb{N}^+$

$$\psi^{<m>} = [n \in \mathbb{N}^+]\Psi(\psi,m,n) \ ,$$

where

$$\Psi(\psi,m,n) = \begin{cases} n & \text{if } n \leq m \\ m + \langle n-m\rangle \cdot \psi & \text{if } n > m \ . \end{cases}$$

It follows that $\varphi_0 = \Theta_0 = [n \in \mathbb{N}^+]n$, the identity on $\mathbb{N}^+$, and that for $\psi \in \mathbb{N}^+ \to \mathbb{N}^+$ we have $\psi^{<0>} = \psi$. Note that φ_m and Θ_m are injective, and that if ψ is injective then so is $\psi^{<m>}$.

Simple computation shows that the following lemmas hold:

<u>LEMMA 2.1</u>. If $k,m \in \mathbb{N}$ then

i. $\varphi_k \circ \varphi_m = \varphi_{k+m}$.

ii. $\Theta_k^{<m>} \circ \Theta_m = \Theta_{k+m}$.

iii. $\Theta_k \circ \varphi_m^{<1>} = \varphi_m \circ \Theta_{k+m}$.

<u>LEMMA 2.2</u>. If $k,m \in \mathbb{N}$ and $\psi \in \mathbb{N}^+ \to \mathbb{N}^+$ then

i. $(\psi^{<k>})^{<m>} = \psi^{<k+m>}$

ii. $\varphi_k \circ \psi^{<k>} = \psi \circ \varphi_k$.

iii. $\Theta_{k-1} \circ \psi^{<k>} = \psi^{<k>} \circ \Theta_{k-1}$.

<u>LEMMA 2.3</u>. If $k \in \mathbb{N}$ and $\psi1,\psi2 \in \mathbb{N}^+ \to \mathbb{N}^+$ then

$$\psi1^{<k>} \circ \psi2^{<k>} = (\psi1 \circ \psi2)^{<k>} \ .$$

LEMMA 2.4. If $k,m \in \mathbb{N}$ and $n \in \mathbb{N}^+$ then

$$\langle n\rangle\varphi_m^{\langle k\rangle} = \begin{cases} n & \text{if } n \leqq k \\ n+m & \text{if } n > k\,. \end{cases}$$

LEMMA 2.5. If $k,\ell,m \in \mathbb{N}$ then

$$\varphi_\ell \circ \varphi_m^{\langle k\rangle} = \begin{cases} \varphi_{\ell+m} & \text{if } k \leqq \ell \\ \varphi_m^{\langle k-\ell\rangle} \circ \varphi_\ell & \text{if } k > \ell\,. \end{cases}$$

3. Terms, transformation and substitution

We define the set of terms Δ inductively as follows:

DEFINITION 3.1.

1. $\tau \in \Delta$
2. if $n \in \mathbb{N}$ then $\underline{n} \in \Delta$
3. if $u,v \in \Delta$ then $\langle u\rangle v \in \Delta$
4. if $u,v \in \Delta$ then $[u]v \in \Delta$.

Transformation, i.e. adaptation of the references in terms by means of a function ψ is defined as follows:

DEFINITION 3.2. Let $\psi \in \mathbb{N}^+ \to \mathbb{N}^+$. Then $\underline{\psi}$ is defined by

1. $\underline{\psi}\tau = \tau$
2. $\underline{\psi}\,\underline{n} = \underline{\langle n\rangle\psi}$
3. $\underline{\psi}\langle u\rangle v = \langle\underline{\psi}u\rangle\underline{\psi}v$
4. $\underline{\psi}[u]v = [\underline{\psi}u]\underline{\psi^{\langle 1\rangle}}v$.

Clearly if $u \in \Delta$ then $\underline{\psi} u \in \Delta$. Moreover

$$\underline{\psi} u = \tau \quad \text{iff} \quad u = \tau,$$

$$\underline{\psi} u = \underline{m} \quad \text{iff} \quad u = \underline{n} \text{ and } \langle n \rangle = m\ ,$$

$$\underline{\psi} u = \langle v1 \rangle v2 \text{ iff } u = \langle u1 \rangle u2,\ \underline{\psi} u1 = v1 \text{ and } \underline{\psi} u2 = v2\ ,$$

$$\underline{\psi} u = [v1]v2 \text{ iff } u = [u1]u2,\ \underline{\psi} u1 = v1 \text{ and } \underline{\psi^{\langle 1 \rangle}} u2 = v2\ .$$

It follows that for injective ψ, $\underline{\psi} u = \underline{\psi} v$ implies $u = v$.

<u>LEMMA 3.1</u>. If $\psi 1, \psi 2 \in \mathbb{N}^+ \to \mathbb{N}^+$, $u \in \Delta$ then

$$\underline{\psi 1}\ \underline{\psi 2} u = \underline{\psi 2 \circ \psi 1} u\ .$$

<u>Proof</u>: By induction on u.

For $u,v \in \Delta$, $k \in \mathbb{N}^+$ we define substitution of u in v at k, denoted by $\sum_k^u v$ as follows:

<u>DEFINITION 3.3</u>.

1. $\sum_k^u \tau = \tau$

2. $$\sum_k^u \underline{n} = \begin{cases} \underline{n} & \text{if } n < k \\ \underline{\varphi_{k-1}} u & \text{if } n = k \\ \underline{n-1} & \text{if } n > k \end{cases}$$

3. $\sum_k^u \langle v1 \rangle v2 = \langle \sum_k^u v1 \rangle \sum_k^u v2$

4. $\sum_k^u [v1]v2 = [\sum_k^u v1] \sum_{k+1}^u v2$.

Clearly, again, if $u,v \in \Delta$ then $\sum_k^u v \in \Delta$.

Now we have the following technical lemmas:

<u>LEMMA 3.2</u>. $\sum_k^u v = \sum_1^{\underline{\varphi_{k-1}} u} \underline{\theta_{k-1}} v$.

Proof: By induction on v.

LEMMA 3.3. $\underline{\psi}\ \sum_1^u v = \sum_1^{\psi u} \underline{\psi^{<1>}} v$

Proof: By induction on v.

LEMMA 3.4. If m < k then $\sum_k^u \underline{\varphi_m} v = \underline{\varphi_m} \sum_{k-m}^u v$

Proof: By Lemma 3.2 and Lemma 2.1.

LEMMA 3.5. If m+ℓ ≧ k > ℓ then $\sum_k^u \underline{\varphi_m^{<\ell>}} v = \underline{\varphi_{m-1}^{<\ell>}} v$

Proof: By induction on v.

COROLLARY 3.5. If m ≧ k then $\sum_k^u \underline{\varphi_m} v = \underline{\varphi_{m-1}} v$.

These lemmas are used to prove the following theorem:

THEOREM 3.1 Substitution theorem.

If m ≧ k then $\sum_m^u \sum_k^v w = \sum_k^{\sum_{m-k+1}^u v} \sum_{m+1}^u w$.

Proof: By induction on w.

The relevant case is when w = $\underline{n}$.

If n= k then

$$\sum_m^u \sum_k^v w = \sum_m^u \underline{\varphi_{k-1}} v = \underline{\varphi_{k-1}} \sum_{m-k+1}^u v \text{ by Lemma 3.4}$$

and on the other hand

$$\sum_k^{\Sigma^u_{m-k+1} v} \sum^u_{m+1} w = \sum_k^{\Sigma^u_{m-k+1} v} \underline{k} = \underline{\varphi_{k-1}} \sum^u_{m-k+1} v \quad .$$

If $n = m+1$ then

$$\sum^u_m \sum^v_k w = \sum^u_m \underline{m} = \underline{\varphi_{m-1}} u$$

and

$$\sum_k^{\Sigma^u_{m-k+1} v} \sum^u_{m+1} w = \sum_k^{\Sigma^u_{m-k+1} v} \underline{\varphi_m} u = \underline{\varphi_{m-1}} u \text{ by Lemma 3.5.}$$

For other values of n the proof is straightforward.

4. Reduction

We define on Δ the relation $\succ$, called <u>one step reduction</u>.

<u>DEFINITION 4.1.</u> 1. $\langle u\rangle[w]v \succ \sum^u_1 v$

If $u \succ v$ then

2. $\langle u\rangle w \succ \langle v\rangle w$

3. $\langle w\rangle u \succ \langle w\rangle v$

4. $[u]w \succ [v]w$

5. $[w]u \succ [w]v$.

The relation $\succcurlyeq$ on Δ is the reflexive and transitive closure of $\succ$, defined by

<u>DEFINITION 4.2.</u> 1. $u \succcurlyeq u$

2. If $u \succcurlyeq v$ and $v \succ w$ then $u \succcurlyeq w$.

It is easily seen that the relation $\succ$ is transitive and monotonic. By induction on $u \succ v$, respectively $\underline{\psi}u \succ \underline{\psi}v$ the following technical lemma is proved:

<u>LEMMA 4.1</u>. If $u \succ v$ then for any ψ $\underline{\psi}u \succ \underline{\psi}v$;

if ψ is injective then $\underline{\psi}u \succ \underline{\psi}v$ implies $u \succ v$.

Another technical lemma:

<u>LEMMA 4.2</u>. If $\underline{\psi}u \succ v$ then for some w $v = \underline{\psi}w$ and $u \succ w$.

Proof: By induction on $\underline{\psi}u \succ v$.

Finally is is easily shown that if $[u1]u2 \succ v$ the $v = [v1]v2$, $u1 \succ v1$ and $u2 \succ v2$.

5. The Church-Rosser theorem

We define on Δ the relation $\supset$ called <u>nested one step reduction</u>.

<u>DEFINITION 5.1</u>. 1. $u \supset u$

If $u \supset u1$ and $v \supset v1$ then

2. $\langle u\rangle[w]v \supset \Sigma_1^{u1}\, v1$

3. $\langle u\rangle v \supset \langle u1\rangle v1$

4. $[u]v \supset [u1]v1$.

$\supseteq$ denotes the transitive (and - of course - reflexive) closure of $\supset$. By an easy inductive argument it is seen that $u \supseteq v$ iff $u \succ v$.

The following technical lemma is proved by induction on $u \supset v$.

LEMMA 5.1. If $u \supset v$ then for any ψ $\underline{\psi}u \supset \underline{\psi}v$.

Now we are able to prove two lemmas on substitution.

LEMMA 5.2. If $u \supset u1$ then $\sum_k^u v \supset \sum_k^{u1} v$.

Proof: By induction on v it is proved that for any k $\sum_k^u v \supset \sum_k^{u1} v$.

LEMMA 5.3. Substitution lemma for $\supset$.

If $u \supset u1$ and $v \supset v1$ then $\sum_k^u v \supset \sum_k^{u1} v1$.

Proof: By induction on $v \supset v1$ it is proved that for any k $\sum_k^u v \supset \sum_k^{u1} v1$.

Lemma 5.2 and Theorem 3.1 are used.

Using these lemmas we can prove the diamond property for $\supset$.

LEMMA 5.4. Diamond lemma for $\supset$.

If $u \supset u1$ and $u \supset u2$ then there exists a term v such that $u1 \supset v$ and $u2 \supset v$.

Proof: By induction on $u \supset u1$ and $u \supset u2$, using Lemma 5.3.

As a corollary we have:

THEOREM 5.1. Church-Rosser theorem for $\geq$.

If $u \geq u1$ and $u \geq u2$ then there exists a term v such that $u1 \geq v$ and $u2 \geq v$.

6. Norms, norming functionals and monotonic functionals

A term $u \in \Delta$ is called normal if $u \geq v$ implies $u = v$. A reduction sequence of u is a finite or infinite sequence $u_0, u_1, u_2, \ldots$ such that $u_0 = u$ and $u_{n-1} > u_n$ for $n \in \mathbb{N}^+$. We say that u strongly normalizes if all reduction sequences of u are finite. This is the case, by König's lemma, iff there is a uniform upperbound to the lengths of the reduction sequences of u.

We will prove strong normalization for a subset of Δ, the set of normable terms. Our proof extends proofs of Gandy [4] and the Vrijer [7] for strong normalization in simple type theory. It is based mainly on de Vrijer's "quick proof" ; we refer also to that proof for comments.

We define the set F of norms recursively as follows:

DEFINITION 6.1. 1. $\mathbb{N} \in F$

2. if $\alpha,\beta \in F$ then $\alpha \to \beta := (\alpha \to \beta) \times \mathbb{N} \in F$.

It is clear that, for $\alpha,\beta \in F$, $\alpha = \beta$ or $\alpha \cap \beta = \emptyset$.

The elements of $\cup F$ will be called norming functionals. For any norming functional f the norm to which f belongs is denoted by $f^{\uparrow}$. Moreover, we define the projection operators:

if $f = n$, $n \in \mathbb{N}$ then $f^* = n$,

if $f = \lceil g,n \rceil$, $\lceil g,n \rceil \in \alpha \to \beta$ then $f' = g$ and $f^* = n$.

Let f be a norming functional, m a natural number. We define the norming functional f + m as follows:

DEFINITION 6.2. 1. If $f \in \mathbb{N}$, $f = n$ then $f + m = n + m$.

2. If $f \in \alpha \to \beta$, $f = \lceil g,n \rceil$

then $f + m = \lceil [h \in \alpha](\langle h \rangle g + m),\ n + m \rceil$.

Thus for $f \in \alpha$ we have $f + m \in \alpha$ and

$(f + m)^* = f^* + m$,

$\langle h \rangle (f + m)' = \langle h \rangle f' + m$ if $\alpha = \beta \to \gamma$ and $h \in \beta$.

Note that + extends addition on the natural numbers.

For $\alpha \in F$ and $n \in \mathbb{N}$ we define the norming functional $c_n^{\alpha} \in \alpha$.

DEFINITION 6.3. 1. $c_n^{\mathbb{N}} = n$

2. $c_n^{\beta\to\gamma} = \lceil [h \in \beta] c^{\gamma}_{h^*+n}, n \rceil$

Thus

$$c_n^{\alpha}{}^* = n,$$

$$\langle h\rangle (c^{\beta\to\gamma})' = c^{\gamma}_{h^*+n} \quad \text{if } h \in \beta.$$

Note that $c_n^{\alpha} + m = c_{n+m}^{\alpha}$.

Now let α be a norm. We define a subset α^0 of α and a relation $<$ on α^0 by a simultaneous inductive definition.

DEFINITION 6.4. 1. $\mathbb{N}^0 = \mathbb{N}$; for $f,g \in \mathbb{N}^0$, $f < g$ iff $f^* < g^*$

2. $(\beta \to \gamma)^0 = \{f \in \beta \to \gamma \mid \forall_{g\in\beta^0}(\langle g\rangle f' \in \gamma^0) \wedge \forall_{g,h\in\beta^0}(g<h \Rightarrow \langle g\rangle f' < \langle h\rangle f')\}$;

for $f,g \in (\beta \to \gamma)^0$, $f < g$ iff $\forall_{h\in\beta^0}(\langle h\rangle f' < \langle h\rangle g') \wedge f^* < g^*$.

We define $G := \{\alpha^0 \mid \alpha \in F\}$; the elements of G will be called monotonic functionals.

Note that $<$ on $\mathbb{N}^0$ is the order on the naturals.

The following facts are easily proved:

If $f,g,h \in \alpha^0$, $f < g$ and $g < h$ then $f < h$.

If $f,g \in \alpha^0$, $m \in \mathbb{N}$ then $f + m \in \alpha^0$ and if $f < g$ then $f+m < g+m$.

If $f \in \alpha^0$, $m,n \in N$ and $m < n$ then $f+m < f+n$.

Moreover

$c_n^{\alpha} \in \alpha^0$ and if $m < n$ then $c_m^{\alpha} < c_n^{\alpha}$.

7. Strong normalization

We will assign to certain terms $u \in \Delta$ a functional in UF, which will be called the norming functional of u. In order to define it we need a sequence $\Phi \in (UF)^*$;

Φ may be thought of as an administration of the functionals assigned to the free variables of u. $fn(u,\Phi)$ will denote the norming functional of u. It may be the case that $fn(u,\Phi)$ is undefined. This will be denoted by $fn(u,\Phi) = \square$. Terms u for which $fn(u,\Phi) \neq \square$ for some $\Phi \in (UG)^*$ will be called <u>normable</u>.

<u>DEFINITION 7.1.</u>

1. $fn(\tau,\Phi) = 0$

2. $$fn(n,\Phi) = \begin{cases} \langle n\rangle\Phi & \text{if } n \leqq L(\Phi) \\ \square & \text{otherwise} \end{cases}$$

3. $$fn(\langle v\rangle w,\Phi) = \begin{cases} \langle fn(v,\Phi)\rangle fn(w,\Phi)' & \text{if } fn(v,\Phi) \neq \square,\ fn(w,\Phi) \neq \square \text{ and } dom(fn(w,\Phi)') = fn(v,\Phi)\uparrow \\ \square & \text{otherwise} \end{cases}$$

4. $$fn([v]w,\Phi) = \begin{cases} [[h \in \alpha]fn(w,h\&\Phi)+h^*+fn(v,\Phi)^*+1,\ fn(v,\Phi)^*+fn(w,c_0^\alpha\&\Phi)^*] \\ \qquad \text{if } fn(v,\Phi) \neq \square,\ fn(v,\Phi)\uparrow = \alpha \text{ and } fn(w,h\&\Phi) \neq \square \text{ for } h \in \alpha \\ \square \qquad \text{otherwise.} \end{cases}$$

It will be clear from Lemma 7.5, which will be proved presently, that for normable terms u $fn(u,\Phi)^*$ is an upperbound for the lengths of the reduction sequences of u.

Note that if $fn(\langle u\rangle[w]v,\Phi) \neq \square$ then $fn(u,\Phi)\uparrow = fn(w,\Phi)\uparrow$.

Our first lemma expresses that it only depends on the norms of the functionals in Φ whether $fn(u,\Phi)$ is defined and, if so, what is the value of $fn(u,\Phi)\uparrow$.

LEMMA 7.1. If $\Phi 1, \Phi 2 \in (UF)^*$, $L(\Phi 1) = L(\Phi 2) = n$ and $\langle k\rangle\Phi 1\uparrow = \langle k\rangle\Phi 2\uparrow$ for $k \leq n$ then either $fn(u,\Phi 1) = fn(u,\Phi 2) = \Box$,

or $\quad fn(u,\Phi 1)\uparrow = fn(u,\Phi 2)\uparrow$.

Proof: By induction on u.

The following technical lemma is also proved by induction on u.

LEMMA 7.2. If $\Phi \in (UF)^*$, $\psi \in \mathbb{N}^+ \to \mathbb{N}^+$ and $\psi \circ \Phi \in (UF)^*$ then

$$fn(\underline{\psi}u,\Phi) = fn(u,\psi \circ \Phi).$$

(Note that Φ as well as ψ is a function, hence $\psi \circ \Phi$ is a function.)

The following important lemma expresses that an upperbound for the lenghts of the reduction sequences of $\sum_1^u v$ can be calculated from $fn(u,\Phi)$ and $fn(v,fn(u,\Phi)\&\Phi)$.

LEMMA 7.3. Substitution lemma.

If $fn(u,\Phi) \neq \Box$ then $fn(\sum_1^u v,\Phi) = fn(v,fn(u,\Phi)\&\Phi)$.

Proof: By induction on v.

The main case is: $v = [v1]v2$.

$$fn(\textstyle\sum_1^u v,\Phi) = \lceil [h \in \alpha]fn(\sum_2^u v2,h\&\Phi)+h*+fn(\sum_1^u v1,\Phi)*+1,\ fn(\sum_1^u v1,\Phi)*+fn(\sum_2^u v2,c_0^\alpha\&\Phi)*\rceil$$

where $\alpha = fn(\sum_1^u v1,\Phi)\uparrow$, while by the induction hypothesis

$$fn(\textstyle\sum_1^u v1,\Phi)\uparrow = fn(v1,fn(u,\Phi)\&\Phi)\uparrow .$$

Moreover, we have by the induction hypothesis for $h \in \alpha$:

$$fn(\textstyle\sum_2^u v2,h\&\alpha) = fn(\sum_1^{\varphi_1 u} \underline{\Theta_1} v2,h\&\Phi) = fn(\underline{\Theta_1} v_2,fn(\underline{\varphi_1} u,h\&\Phi)\&h\&\Phi).$$

Therefore

$$fn(\sum_2^u v2,h\&\alpha) = fn(v2,\Theta_2\circ(fn(u,\Phi)\&h\&\Phi)) = fn(v2,h\&fn(u,\Phi)\&\Phi).$$

It follows that

$$fn(\sum_1^u v,\Phi) = fn(v,fn(u,\Phi)\&\Phi).$$

In order to formulate the next lemma we need the concept of a free variable. Therefore we define for $u \in \Delta$ and $k \in \mathbb{N}^+$ the proposition free(u,k), expressing (in the language of section 1) that the term u contains a reference (or an arrow) to the k-th binding node below u.

DEFINITION 7.2.

1. not free(τ,k)
2. free($\underline{n}$,k) iff n = k
3. free(<v>w,k) iff free(v,k) or free(w,k)
4. free([v]w,k) iff free(v,k) or free(w,k+1).

LEMMA 7.4. Monotonicity lemma.

If $\Phi \in (UG)^*$ then $fn(u,\Phi) \in (UG) \cup \{\square\}$.

If $\Phi1,\Phi2 \in (UG)^*$, $L(\Phi1) = L(\Phi2) = n$, $\langle k\rangle\Phi1 < \langle k\rangle\Phi2$

and for $\ell \leq n$, $\ell \neq k$ $\langle\ell\rangle\Phi1 = \langle\ell\rangle\Phi2$

then $fn(u,\Phi1) < fn(u,\Phi2)$ or $fn(u,\Phi1) = fn(u,\Phi2) = \square$ if free(u,k)

and $fn(u,\Phi1) = fn(u,\Phi2)$ if not free(u,k).

Proof: By induction on u.

The main case is, again, u = [u1]u2.

Suppose $fn(u,\Phi) \neq \square$. Then by the induction hypothesis $fn(u1,\Phi) \in UG$.

Let α denote fn(u1,Φ)$\uparrow$. Then also by the induction hypothesis for every $g \in \alpha$ we have fn(u2,g&Φ) $\in$ UG.

Now let g,h be elements of α such that g < h.

Then either fn(u2,g&Φ) < fn(u2,h&Φ) or fn(u2,g&Φ) = fn(u2,h&Φ), hence fn(u2,g&Φ)+g*+fn(u1,Φ)*+1 < fn(u2,h&Φ)+h*+fn(u1,Φ)*+1.

It follows that fn(u,Φ) $\in$ UG.

Now assume that free(u,k). Then for $g \in \alpha$ we have:

$$\langle g\rangle fn(u,\Phi 1)' = fn(u2,g\&\Phi 1)+g^*+fn(u1,\Phi 1)^*+1$$

and

$$\langle g\rangle fn(u,\Phi 2)' = fn(u2,g\&\Phi 2)+g^*+fn(u1,\Phi 2)^*+1$$

and therefore

$$\langle g\rangle fn(u,\Phi 1)' < \langle g\rangle fn(u,\Phi 2)' .$$

Moreover

$$fn((u,\Phi 1)^* = fn(u1,\Phi 1)^* + fn(u2,c_0^\alpha\&\Phi 1)^*$$

and

$$fn(u,\Phi 2)^* = fn(u1,\Phi 2)^*+fn(u2,c_0^\alpha\&\Phi 2)^*$$

and therefore

$$fn(u,\Phi 1)^* < fn(u,\Phi 2)^*.$$

Hence if free(u,k) then fn(u,Φ1) < fn(u,Φ2).

It is easily seen that if not free(u,k) then fn(u,Φ1) = fn(u,Φ2).

<u>LEMMA 7.5. Reduction lemma.</u>

If $\Phi \in (UG)^*$, fn(u,Φ) $\neq \square$ then u $\succ$ v implies fn(v,Φ) < fn(u,Φ).

<u>Proof</u>: By induction on u $\succ$ v. The case u = <u1>[u3]u2, v = $\sum_1^{u_1}$u2 is covered by Lemma 7.3.

As a corollary we have

THEOREM 7.1. Strong normalization.

If u is normable then u strongly normalizes.
If $\Phi \in (UG)^*$, $fn(u,\Phi) \neq \Box$ then $fn(u,\Phi)^*$ is an upperbound for the lengths of reduction sequences of u.

8. Contexts and types

In Sections 8 and 9 we will define the system Λ_∞. In order to do so we must be able to calculate the type of an expression $u \in \Delta$. For assigning a type to u we need a sequence $U \in \Delta^*$. Such a sequence is called a context. It can be considered as administrating the types of the free variables in u. The type of u may be undefined which, again, will be denoted by the symbol "□".

DEFINITION 8.1,

1. $typ(\tau,U) = \Box$

2. $typ(\underline{n},U) = \begin{cases} \underline{\varphi_n}\langle n\rangle U & \text{if } n \leq L(U) \\ \Box & \text{otherwise} \end{cases}$

3. $typ(\langle v\rangle w,U) = \begin{cases} \langle v\rangle typ(w,U) & \text{if } typ(w,U) \neq \Box \\ \Box & \text{otherwise} \end{cases}$

4. $typ([v]w,U) = \begin{cases} [v]typ(w,v\&U) & \text{if } typ(w,v\&U) \neq \Box \\ \Box & \text{otherwise} \end{cases}$

In order to express the properties of the typing operator typ, we must extend the transformation operation, the substitution operation and the reduction relation to contexts. As far as transformation is concerned we restrict ourselves to the functions $\underline{\varphi_m}^{\langle k\rangle}$.

DEFINITION 8.2. Let U be a context, L(U) = n.

Then

$\underline{\varphi_m}^{\langle k\rangle} U \in \Delta^*$ with $L(\underline{\varphi_m}^{\langle k\rangle} U) = n$ is defined by

$$\langle \ell \rangle \underline{\varphi_m}^{k} U = \begin{cases} \underline{\varphi_m}^{\langle k-\ell\rangle} \langle \ell \rangle U & \text{if } \ell \leqq k,\ \ell \leqq n \\ \langle \ell \rangle U & \text{if } k < \ell \leqq n\ . \end{cases}$$

The following lemmas are easily seen to hold:

LEMMA 8.1. $\underline{\varphi_m}^{\langle 0\rangle} U = U$; $\underline{\varphi_m}^{\langle k+1\rangle}(u\&U) = \underline{\varphi_m}^{\langle k\rangle} u \ \&\ \underline{\varphi_m}^{\langle k\rangle} U$

LEMMA 8.2. If L(U1) = k then $\underline{\varphi_m}^{\langle k\rangle}(U1\&U2) = (\underline{\varphi_m}^{\langle k\rangle} U1)\&U2$.

We prove a technical lemma by induction on u:

LEMMA 8.3. If L(U0) = k, L(U1) = m and U = U0&U1&U2

then either $typ(\underline{\varphi_m}^{\langle k\rangle} u, \underline{\varphi_m}^{\langle k\rangle} U) = typ(u, U0\&U2) = \square$

or $typ(\underline{\varphi_m}^{\langle k\rangle} u, \underline{\varphi_m}^{\langle k\rangle} U) = \underline{\varphi_m}^{\langle k\rangle} typ(u, U0\&U2)$.

This gives as a consequence:

COROLLARY 8.3. If L(U1) = m, then either

$$typ(\underline{\varphi_m} u, U1\&U2) = typ(u, U2) = \square$$

or

$$typ(\underline{\varphi_m} u, U1\&U2) = \underline{\varphi_m} typ(u, U2)\ .$$

Now in order to investigate the relation between substitution and typing we define substitution in contexts:

<u>DEFINITION 8.3</u>. Let U be a context, L(U) = n, and $1 \leqq k \leqq n$.
Then $\Sigma^u_k U \in \Delta^*$ with $L(\Sigma^u_k U) = n-1$ is defined by

$$\langle \ell \rangle \, \Sigma^u_k U = \begin{cases} \Sigma^u_{k-\ell} \langle \ell \rangle U & \text{if } \ell < k \\ \\ \langle \ell + 1 \rangle U & \text{if } k \leqq \ell < n. \end{cases}$$

We have the following easy lemmas on substitution in contexts:

<u>LEMMA 8.4</u>. $\Sigma^u_1 (v \& U) = U$; $\Sigma^u_{k+1} (v \& U) = \Sigma^u_k v \;\&\; \Sigma^u_k U$.

<u>LEMMA 8.5</u>. If L(U1) = k then $\Sigma^u_k (U1 \& U2) = (\Sigma^u_k U1) \& U2$.

The next lemma describes the relation between substitution and typing:

<u>LEMMA 8.6. Substitution lemma for typ</u>.

If $typ(\underline{\varphi_k} u, U) \succ w$ and $\underline{\varphi_k} \langle k \rangle U \succ w$

then either $typ(\Sigma^u_k v, \Sigma^u_k U) = typ(v, U) = \square$

or

$$typ(\Sigma^u_k v, \Sigma^u_k U) \succ w0 \text{ and } \Sigma^u_k typ(v, U) \succ w0 \text{ for some } w0 \in \Delta.$$

<u>Proof</u>: By induction on v. The main case is $v = \underline{k}$.

Because $k \leqq L(U)$ we have $U = U1\&U2$, where $L(U1) = k$.

Therefore $\mathrm{typ}(\sum_k^u v, \sum_k^u U) = \mathrm{typ}(\underline{\varphi_{k-1}}u, (\sum_k^u U1)\&U2)$ where $L(\sum_k^u U1) = k-1$.

Hence, by Corollary 8.3 : $\mathrm{typ}(\sum_k^u v, \sum_k^u U) = \underline{\varphi_{k-1}}\mathrm{typ}(u, U2)$,

so, again by Corollary 8.3:

$$\underline{\varphi_1}\mathrm{typ}(\sum_k^u v, \sum_k^u U) = \underline{\varphi_k}\mathrm{typ}(u, U2) = \mathrm{typ}(\underline{\varphi_k}u, U) \succ w.$$

On the other hand

$$\sum_k^u \mathrm{typ}(v, U) = \sum_k^u \underline{\varphi_k}\langle k\rangle U = \underline{\varphi_{k-1}}\langle k\rangle U \text{ by Corollary 3.5.}$$

This gives us

$$\underline{\varphi_1} \sum_k^u \mathrm{typ}(v, U) = \underline{\varphi_k}\langle k\rangle U \succ w$$

By Lemma 4.2 it follows that $w = \underline{\varphi_1}\, w0$ and that

$$\mathrm{typ}(\sum_k^u v, \sum_k^u U) \succ w0 \text{ and } \sum_k^u \mathrm{typ}(v, U) \succ w0.$$

<u>COROLLARY 8.6</u>. If $\mathrm{typ}(u, V) \succ w$ and $v1 \succ w$

then either

$$\mathrm{typ}(\sum_1^u v, V) = \mathrm{typ}(v, v1\&V) = \square$$

or

$$\mathrm{typ}(\sum_1^u v, V) \succ w0 \quad \text{and } \sum_1^u \mathrm{typ}(v, v1\&V) \succ w0 \text{ for some } w0 \in \Delta.$$

<u>Proof</u>: Take $k = 1$ and $U = v1\&V$ in Lemma 8.6.

Finally, in order to describe a relation between typing and reduction we define the concept of reduction on contexts.

DEFINITION 8.4. Let u and v be terms, U and V contexts.

1. if $u \succ v$ then $u\&U \succ v\&U$

2. if $U \succ V$ then $u\&U \succ u\&V$.

We have the following lemma:

LEMMA 8.7. If $U \succ V$ then $L(U) = L(V) = n > 0$ and there is just one $k \leq n$ such that

$$<k>U \succ <k>V \quad \text{and} \quad <\ell>U = <\ell>V \quad \text{for } \ell \leq n,\ \ell \neq k.$$

Proof: By induction on $U \succ V$.

Moreover we have

LEMMA 8.8. If $U \succ V$ then either $typ(u,U) \succ typ(u,V)$

or $typ(u,U) = typ(u,V)$

Proof: By induction on u.

COROLLARY 8.8. If $v \succ w$ then either $typ(u,v\&U) \succ typ(u,w\&U)$

or $typ(u,v\&U) = typ(u,w\&U)$.

The relatiom $\succeq$ between contexts is the reflexive and transitive closure of $\succ$. If $u \succeq v$ and $U \succeq V$ then clearly $u\&U \succeq v\&V$.

9. The system Λ_∞

We will define by simultaneous induction the set $\Gamma_\infty \subset \Delta^*$ which is the set of correct contexts, and the set $\Lambda_\infty \subset \Delta \times \Delta^*$. (it will turn out even $\Lambda_\infty \subset \Delta \times \Gamma_\infty$). If $\lceil u,U \rceil \in \Lambda_\infty$ u will be called a correct term on context U. Here correctness should be understood as follows:

If <u>v is correct on context U then v "is a function" and moreover typ(u,U) and "the domain of v" have a common reduct. In fact, we have not formalized what it means for v to "be a function" and, if it is, what "the domain of v" is. The requirements described above appear however in clause 4 of our definition and - implicitly - also in clause 6.

Together with Γ_∞ and Λ_∞ we will define the sets Γ_i and Λ_i for $i \in \mathbb{N}$. They are introduced only for the purpose of induction in the proof of Lemma 10.3. If $\lceil u,U \rceil \in \Lambda_i$ then u will be called <u>i-correct</u>. The systems are connected with the notion of <u>degree</u> in [2] and [3] in the sense that any i-correct term will have degree at most i. (The converse however does not hold.)

In the following discussion it is always assumed that $i \in \mathbb{N}^\infty$. For $i = \infty$ the definitions and lemmas contain the theory of Λ_∞.

<u>DEFINITION 9.1.</u> 0. $\Gamma_0 = \Lambda_0 = \emptyset$

If $i > 0$ then

1. $\emptyset \in \Gamma_i$
2. if $\lceil u,U \rceil \in \Lambda_i$ then $u\&U \in \Gamma_i$
3. if $U \in \Gamma_i$ then $\lceil \tau,U \rceil \in \Lambda_i$
4. if $\mathrm{typ}(\langle u\rangle v,U) = \square$, $\lceil u,U \rceil \in \Lambda_i$, $\lceil v,U \rceil \in \Lambda_i$, $\mathrm{typ}(u,U) \succ v1$ and $v \succ [v1]v2$ then $\lceil \langle u\rangle v,U \rceil \in \Lambda_i$
5. if $\mathrm{typ}([u]v,U) = \blacksquare$ and $\lceil v,u\&U \rceil \in \Lambda_i$ then $\lceil [u]v,U \rceil \in \Lambda_i$
6. if $\lceil \mathrm{typ}(u,U),U \rceil \in \Lambda_{i-1}$ then $\lceil u,U \rceil \in \Lambda_i$.

Clearly if $\lceil u,U \rceil \in \Lambda_i$ then $U \in \Gamma_i$ and if $U1\&U2 \in \Gamma_i$ then $U2 \in \Gamma_i$. It is also clear (by induction on i) that $\Lambda_i \subset \Lambda_{i+1}$ for $i \in \mathbb{N}$ and it is easy to check that $\Lambda_\infty = \bigcup\{\Lambda_i \mid i \in \mathbb{N}\}$.

We have the following technical lemma:

<u>LEMMA 9.1</u>. If L(U0) = k, L(U1) = m, U = U0&U1&U2 and U1&U2 $\in \Gamma_i$ then

$$\lceil \underline{\varphi}_m^{<k>} u, \underline{\varphi}_m^{<k>} U \rceil \in \Lambda_i \text{ iff } \lceil u, U0\&U2 \rceil \in \Lambda_i.$$

<u>Proof</u>: By induction, respectively on $\lceil u, U0\&U2 \rceil \in \Lambda_i$ and on $\lceil \underline{\varphi}_m^{<k>} u, \underline{\varphi}_m^{<k>} U \rceil \in \Lambda_i$, where frequent use is made of Lemma 8.3.

The lemma has some nice corollaries:

<u>COROLLARY 9.1.1. Weakening and strengthening lemma</u>.

If L(U1) = m, U1&U2 $\in \Gamma_i$, then $\lceil \underline{\varphi}_m u, U1\&U2 \rceil \in \Lambda_i$ iff $\lceil u, U2 \rceil \in \Lambda_i$.

<u>COROLLARY 9.1.2</u>. If U $\in \Gamma_i$, k < L(U) then $\lceil \underline{\varphi}_k <k> U, U \rceil \in \Lambda_i$.

<u>COROLLARY 9.1.3</u>. $\lceil \underline{n}, U \rceil \in \Lambda_\infty$ iff U $\in \Gamma_\infty$ and n $\leq$ L(U).

The next lemma partially expresses our assertions about correctness of terms.

<u>LEMMA 9.2. Soundness of application</u>.

If $\lceil <u>[w]v, U \rceil \in \Lambda_i$ then typ(u,U) $\succ$ w0 and w $\succ$ w0 for some w0 $\in \Delta$.

<u>Proof</u>: By induction on $\lceil <u>[w]v, U \rceil \in \Lambda_i$.

Types of correct terms are, in a sense, preserved under reduction.

<u>LEMMA 9.3. Preservation of types</u>.

If $\lceil u, U \rceil \in \Lambda_i$, u > v then either typ(u,U) = typ(v,U) = $\square$

or typ(u,U) $\succ$ w and typ(v,U) $\succ$ w for some w $\in \Delta$.

Proof: By induction on $u \succ v$.

We will consider the case $u = \langle u1\rangle[u3]u2$, $v = \sum_1^{u1} u2$.

By the previous lemma $typ(u1,U) \succ w0$ and $u3 \succ w0$.

Now $typ(u,U) = \langle u1\rangle[u3]typ(u2,u3\&U) \succ \sum_1^{u1} typ(u2,u3\&U)$

and $typ(v,U) = typ(\sum_1^{u1} u2,U)$. Apply Corollary 8.6.

The following lemmas are easy to prove. The first contains the converse of clause 6 in Definition 9.1.

LEMMA 9.4. Correctness of types.

It $typ(u,U) \neq \square$ then $\lceil u,U\rceil \in \Lambda_i$ iff $\lceil typ(u,U),U\rceil \in \Lambda_{i-1}$.

The second tells us that if an application of a function to an argument is correct, then both the function and the argument are correct.

LEMMA 9.5. Correctness of functions and arguments.

If $\lceil \langle u\rangle v,U\rceil \in \Lambda_i$ then $\lceil u,U\rceil \in \Lambda_i$ and $\lceil v,U\rceil \in \Lambda_i$.

We prove two lemmas which are, in a sense, converses of Lemma 9.5.

LEMMA 9.6. If $\lceil \langle u\rangle v1,U\rceil \in \Lambda_i$, $\lceil v2,U\rceil \in \Lambda_i$, $v1 \succ w$ and $v2 \succ w$ then $\lceil \langle u\rangle v2,U\rceil \in \Lambda_i$.

Proof: By induction on $\lceil \langle u\rangle v1,U\rceil \in \Lambda_i$.

We consider the case of clause 4:

$typ(\langle u\rangle v1,U) = \square$, $\lceil u,U\rceil \in \Lambda_i$, $\lceil v1,U\rceil \in \Lambda_i$, $typ(u,U) \succ w0$ and $v1 \succ [w0]w1$.

We know that $typ(v1,U) = \square$, hence, by Lemma 9.3 $typ(w,U) = \square$ and also $typ(v2,U) = \square$. Therefore $typ(\langle u\rangle v2,U) = \square$.

Moreover, by the Church-Rosser theorem we have, for some $w2 \in \Delta$:

$$w \geq w2 \text{ and } [w0]w1 \geq w2, \text{ hence } w2 = [w0^*]w1^* \text{ for some } w0^* \text{ and } w1^*.$$

Therefore $typ(u,U) \geq w0 \geq w0^*$ and $v2 \geq w \geq [w0^*]w1^*$, so, by clause 4, $\lceil \langle u\rangle v2,U \rceil \in \Lambda_i$.

<u>LEMMA 9.7</u>. If $\lceil \langle u1\rangle v,U \rceil \in \Lambda_i$, $\lceil u2,U \rceil \in \Lambda_i$ and $u1 \geq u2$ then $\lceil \langle u2\rangle v,U \rceil \in \Lambda_i$.

<u>Proof</u>: By induction $\lceil \langle u1\rangle v,U \rceil \in \Lambda_i$.

We consider again the case of clause 4:

$typ(\langle u1\rangle v,U) = \square$, $\lceil u1,U \rceil \in \Lambda_i$, $\lceil v,U \rceil \in \Lambda_i$, $typ(u1,U) \geq w1$ and $v \geq [w1]w2$.

First we have $typ(v,U) = \square$, hence $typ(\langle u2\rangle v,U) = \square$.

By Lemma 9.3 we have for some $w0$ $typ(u1,U) \geq w0$ and $typ(u2,U) \geq w0$.

Hence, by the Church-Rosser theorem: $w0 \geq v1$ and $w1 \geq v1$ for some $v1$.

Therefore $typ(u2,U) \geq w0 \geq v1$ and $v \geq [w1]w2 \geq [v1]w2$, so, by clause 4, $\lceil \langle u2\rangle v,U \rceil \in \Lambda_i$.

Finally we state a lemma on correct abstraction:

<u>LEMMA 9.8</u>. $\lceil [u]v,U \rceil \in \Lambda_i$ iff $\lceil v,u\&U \rceil \in \Lambda_i$.

<u>Proof</u>: By induction, respectively on $\lceil [u]v, U \rceil \in \Lambda_i$ and on $\lceil v,u\&U \rceil \in \Lambda_i$.

10. <u>Closure for Λ_∞</u>

For the proof that Λ_∞ is closed under reduction we need Lemma 10.2 which tells us that correctness is preserved under correct substitution. In order to prove this lemma we give a slightly different definition of Λ_i,

which we will prove to be equivalent to the first definition. Induction on this alternative definition will be used in the proof of Lemma 10.2.
We define for $i \in \mathbb{N}^{\infty}$ the sets C_i and L_i by a simultaneous inductive definition as follows:

DEFINITION 10.1.

0. $C_0 = L_0 = \emptyset$

If $i > 0$ then

1. $\emptyset \in C_i$
2. If $\lceil u,U \rceil \in L_i$ then $u\&U \in C_i$
3. if $U \in C_i$ then $\lceil \tau,U \rceil \in L_i$
4. if typ(<u>v,U) = □, $\lceil u,U \rceil \in L_i$, $\lceil v,U \rceil \in L_i$, typ(u,U) $\succ$ v1 and v $\succ$ [v1]v2 then $\lceil$<u>v,U$\rceil \in L_i$
5. if typ([u]v,U) = ◘ and $\lceil v,u\&U \rceil \in L_i$ then $\lceil [u]v,U \rceil \in L_i$

6.1. if $\lceil$typ($\underline{n}$,U),U$\rceil \in L_{i-1}$ then $\lceil \underline{n},U \rceil \in L_i$

6.2 if $\lceil$typ(<u>v,U),U$\rceil \in L_{i-1}$ and $\lceil v,U \rceil \in L_i$ then $\lceil$<u>v,U$\rceil \in L_i$

6.3. if $\lceil$typ([u]v,U),U$\rceil \in L_{i-1}$ and $\lceil v,u\&U \rceil \in L_i$ then $\lceil [u]v,U \rceil \in L_i$.

The clauses 0 to 5 are the same as the corresponding clauses of Definition 9.1, but clause 6 of that defintion has been split up into three clauses. We easily verify that $L_{i-1} \subset L_i$ and that $L_\infty = \cup\{L_i | i \in \mathbb{N}\}$. In order to show that $C_i = \Gamma_i$ and $L_i = \Lambda_i$ we first prove the following lemma:

LEMMA 10.1. If $\lceil$typ(u,U),U$\rceil \in L_{i-1}$ then $\lceil u,U \rceil \in L_i$.

Proof: By induction on $\lceil typ(u,U) \rceil \in L_{i-1}$.

We consider the case of clause 4: $typ(u,U) = \langle u1\rangle v$, $typ(\langle u1\rangle v,U) = \square$,

$\lceil u1,U \rceil \in L_{i-1}$, $\lceil v,U \rceil \in L_{i-1}$, $typ(u1,U) \geq w1$, $v \geq [w1]w2$.

Now either $u = \underline{n}$ or $u = \langle u1\rangle u2$ and $typ(u2,U) = v$.

If $u = \underline{n}$ then $\lceil u,U \rceil \in L_i$ by clause 6.1.

If $u = \langle u1\rangle u2$, $typ(u2,U) = v$ then we have by the induction hypothesis

$\lceil u2,U \rceil \in L_i$ and therefore $\lceil u,U \rceil \in L_i$ by clause 6.2.

As another case we consider clause 6.3: $typ(u,U) = [u1]v$,

$\lceil typ([u1]v,U),U \rceil \in L_{i-2}$ and $\lceil v,u1\&U \rceil \in L_{i-1}$.

Again we either have $u = \underline{n}$ or $u = [u1]u2$ and $typ(u2, u1\&U) = v$.

If $u = \underline{n}$ then again clause 6.1 applies.

And if $u = [u1]u2$, $typ(u2,u1\&U) = v$ then by the induction hypothesis

$\lceil u2,u1\&U \rceil \in L_i$ and therefore $\lceil u,U \rceil \in L_i$ by clause 6.3.

COROLLARY 10.1. $C_i = \Gamma_i$ and $L_i = \Lambda_i$.

Proof: $L_i \subset \Lambda_i$ is trivial, $\Lambda_i \subset L_i$ is proved by using Lemma 10.1.

Now we are able to prove the following important substitution lemma.

LEMMA 10.2. Substitution lemma for L_i.

If $\lceil \underline{\varphi_k} u,U \rceil \in L_i$, $\lceil v,U \rceil \in L_i$, $typ(\underline{\varphi_k} u,U) \geq w$ and $\underline{\varphi_k}\langle k\rangle U \geq w$ then

$\lceil \sum_k^u v, \sum_k^u U \rceil \in L_i$.

Proof: By induction on $\lceil v,U \rceil \in L_i$, freely using Corollary 10.1.

We consider some of the clauses:

Clause 3. $v = \tau$. We have to prove the $\sum_k^u U \in C_i$.

If $k = 1$ this is clear by Lemma 8.4.

If $k > 1$ then $U = w\&V$ and $\sum_k^u U = \sum_k^u w \; \& \; \sum_k^u V$, also by Lemma 8.4.

Now we have $\lceil w,V \rceil \in L_i$, hence by the induction hypothesis $\lceil \sum_k^u w, \sum_k^u V \rceil \in L_i$ and therefore $\sum_k^u U \in C_i$ by clause 2.

Clause 4: $v = \langle v1 \rangle v2$.

We know that $typ(v,U) = \square$, $\lceil v1,U \rceil \in L_i$, $\lceil v2,U \rceil \in L_i$,

$$typ(v1,U) \geq w1 \text{ and } v2 \geq [w1]w2.$$

By Lemma 8.6 we have: $typ(\sum_k^u v, \sum_k^u U) = \square$ (i).

The induction hypothesis gives us:

$$\lceil \sum_k^u v1, \sum_k^u U \rceil \in L_i \text{ and } \lceil \sum_k^u v2, \sum_k^u U \rceil \in L_i \qquad \text{(ii).}$$

Also by Lemma 8.6 we see that

$$typ(\sum_k^u v1, \sum_k^u U) \geq w0 \text{ and } \sum_k^u typ(v1,U) \geq w0 \text{ for some } w0.$$

Now by Lemma 5.3 it follows that

$$\sum_k^u typ(v1,U) \geq \sum_k^u w1$$

and also

$$\sum_k^u v2 \geq \sum_k^u [w1]w2 = [\sum_k^u w1] \sum_{k+1}^u w2 \;,$$

hence by the Church-Rosser theorem

$$w0 \geq w \text{ and } \sum_k^u w1 \geq w \text{ for some } w.$$

Therefore we have:

$$typ(\sum_k^u v1, \sum_k^u U) \geq w0 \geq w \text{ and } \sum_k^u v2 \geq [\sum_k^u w1] \sum_{k+1}^u w2 \geq [w] \sum_{k+1}^u w2 \qquad \text{(iii).}$$

From (i), (ii) and (iii) we conclude by clause 4 that

$$\lceil \sum_k^u v, \sum_k^u U \rceil \in L_i.$$

Clause 6.1: $v = \underline{n}$. We know that $\lceil typ(v,U), U \rceil = \lceil \underline{\varphi_n} <n>U, U \rceil \in L_{i-1}$.

We discern two cases: $n = k$ and $n \neq k$.

Suppose $n = k$. As $L(U) \geq k$ we may put $U = U1\&U2$ with $L(U1) = k$.

Then $\sum_k^u U = (\sum_k^u U1)\&U_2$ by Lemma 8.5 and $L(\sum_k^u U1) = k-1$.

Moreover it can be shown, just as under clause 3, that $\sum_k^u U \in C_i$.

Hence by Corollary 9.1.1 we have $\lceil u, U2 \rceil \in L_i$ and by the same corollary also $\lceil \sum_k^u v, \sum_k^u U \rceil = \lceil \underline{\varphi_{k-1}} u, \sum_k^u U \rceil \in L_i$.

Now suppose $n \neq k$. $\sum_k^u v$ either equals $\underline{n}$ (if $n < k$) or $\underline{n-1}$ (if $n > k$).

Using Lemma 3.4 (for $n < k$) or Corollary 3.5 (for $n > k$) we see that

$$typ(\sum_k^u v, \sum_k^u U) = \sum_k^u \underline{\varphi_n} <n>U.$$

By the induction hypothesis we have $\lceil \sum^u \underline{\varphi_n} <n>U, \sum_k^u U \rceil \in L_{i-1}$ and therefore by clause 6.1 $\lceil \sum_k^u v, \sum_k^u U \rceil \in L_i$.

Clause 6.2: $v = <v1>v2$.

We know that $\lceil <v1>typ(v2,U), U \rceil \in L_{i-1}$ and that $\lceil v2, U \rceil \in L_i$.

By the induction hypothesis it follows that

$$\lceil <\sum_k^u v1> \sum_k^u typ(v2,U), \sum_k^u U \rceil \in L_{i-1} \qquad \text{(i)}$$

and

$$\lceil \sum_k^u v2, \sum_k^u U \rceil \in L_i . \qquad (*)$$

By Lemma 8.6 it is known that for some $w0 \in \Delta$

$$\sum_k^u typ(v2,U) \succ w0 \text{ and } typ(\sum_k^u v2, \sum_k^u U) \succ w0. \qquad \text{(ii)}$$

And from (*) we conclude by Lemma 9.4 that

$$\lceil \mathrm{typ}(\textstyle\sum_k^u v2, \sum_k^u U), \sum_k^u U \rceil \in L_{i-1}. \qquad \text{(iii)}$$

From (i), (ii) and (iii) it follows by Lemma 9.6 that

$$\lceil \mathrm{typ}(\textstyle\sum_k^u v, \sum_k^u U), \sum_k^u U \rceil \in L_{i-1} ,$$

and this, together with (*) gives us by clause 6.1:

$$\lceil \textstyle\sum_k^u v, \sum_k^u U \rceil \in L_i .$$

We leave the other clauses to the reader.

<u>COROLLARY 10.2</u>. If $\lceil u,V \rceil \in L_i$, $\lceil v,v1\&V \rceil \in L_i$, $\mathrm{typ}(u,V) \succ w$ and $v1 \succ w$ then $\lceil \sum_1^u v,V \rceil \in L_i$.

<u>Proof</u>: Take k = 1 and U = v1&V in Lemma 10.2.

Our next lemma implies that for $i \in \mathbb{N}$ Λ_i is closed under reduction. In order to word it we use the relation $\succ$ between contexts, which has been defined in section 8.

In order to prove the lemma we assign to every context U the number M(U) which is the sum of the lengths of the terms in U:

if L(U) = n then M(U) = L(<1>U) + L(<2>U) + ... + L(<n>U).

<u>LEMMA 10.3</u>. If $i \in \mathbb{N}$, $u\&U \succ v\&V$ and $\lceil u,U \rceil \in \Lambda_i$ then $\lceil v,V \rceil \in \Lambda_i$.

<u>Proof</u>: By induction on i.

If i = 0 then $\Lambda_i = \emptyset$, so the lemma holds.

Suppose i > 0. We prove the following:

<u>PROPOSITION</u>: If $u\&U \succ v\&V$ and $\lceil u,U \rceil \in \Lambda_i$ then $\lceil v,V \rceil \in \Lambda_i$.

<u>Proof</u>: By induction on $M(u\&U)$.

If $M(u\&U) = 1$ then $u\&U \succ v\&V$ is impossible, so the proposition holds.

Now suppose $M(u\&U) > 1$.

As $u\&U \succ v\&V$ we have either $u \succ v$ and $U = V$ or $u = v$ and $U \succ V$.

Suppose $u \succ v$ and $U = V$. We inspect the clauses for $u \succ v$.

1. $u = \langle u1\rangle[u3]u2$, $v = \Sigma_1^{u1} u2$.

By Lemma 9.2 we have $typ(u1,U) \succ w$ and $u3 \succ w$ for some w, and by Lemma 9.5 $\lceil [u3]u2,U \rceil \in \Lambda_i$ so $\lceil u2,u3\&U \rceil \in \Lambda_i$ by Lemma 9.8.

Apply Corollary 10.2.

2. $u = \langle u1\rangle u2$, $u1 \succ v1$, $v = \langle v1\rangle u2$.

By Lemma 9.5 we have $\lceil u1,U \rceil \in \Lambda_i$.

Moreover $u1\&U \succ v1\&U$ and $M(u1\&U) < M(u\&U)$.

Therefore by our induction hyothesis we have $\lceil v1,U \rceil \in \Lambda_i$ and hence $\lceil v,U \rceil \in \Lambda_i$ by Lemma 9.7.

3. $u = \langle u1\rangle u2$, $u2 \succ v2$, $v = \langle u1\rangle v2$.

$\lceil v,U \rceil \in \Lambda_i$ by a similar argument, where Lemma 9.6 is used instead of Lemma 9.7.

4. $u = [u1]u2$, $u1 \succ v1$, $v = [v1]u2$.

By Lemma 9.8 we have $\lceil u2,u1\&U \rceil \in \Lambda_i$.

Moreover $u2\&u1\&U \succ u2\&v1\&U$ and $M(u2\&u1\&U) < M(u\&U)$;

in fact $M(u\&U) = M(u2\&u1\&U) + 2$.

Therefore our induction hypothesis gives us $\lceil u2,v1\&U \rceil \in \Lambda_i$ and it follows that $\lceil v,U \rceil \in \Lambda_i$ by Lemma 9.8.

5. $u = [u1]u2$, $u2 \succ v2$, $v = [u1]v2$.

$\lceil v,U \rceil \in \Lambda_i$ by a similar argument as under 4.

Now suppose $u = v$ and $U > V$. We inspect the clauses for $\lceil u,U \rceil \in \Lambda_i$.

3. $u = \tau$. We have to prove that $V \in \Gamma_i$.

As $U > V$ it is impossible that $U = \emptyset$, so we may put $U = u1\&U1$ and $V = v1\&V1$.

As $U \in \Gamma_i$ we have $\lceil u1,U1 \rceil \in \Lambda_i$ and also $M(U) < M(u\&U)$.

Therefore we have by our induction hypothesis $\lceil v1,V1 \rceil \in \Lambda_i$, hence $V \in \Gamma_i$.

4. $u = \langle u1 \rangle u2$, $typ(u,U) = \square$, $\lceil u1,U \rceil \in \Lambda_i$, $\lceil u2,U \rceil \in \Lambda_i$,

$typ(u1,U) \geq v1$ and $u2 \geq [v1]v2$.

By Lemma 8.7 we know $typ(u,V) = \square$. Moreover, we have $u1\&U > u1\&V$ and $M(u1\&U) < M(u\&U)$ so by our induction hypothesis $\lceil u1,V \rceil \in \Lambda_i$, and by a similar argument we see that $\lceil u2,V \rceil \in \Lambda_i$.

Also by Lemma 8.7 it is seen that $typ(u1,U) \geq typ(u1,V)$ so by the Church-Rosser theorem we have:

$v1 \geq w$ and $typ(u1,V) \geq w$ for some w.

It follows that $u2 \geq [v1]v2 \geq [w]v2$, hence $\lceil u,V \rceil \in \Lambda_i$ by clause 4.

5. $u = [u1]u2$, $typ(u,U) = \square$, $\lceil u2,u1\&U \rceil \in \Lambda_i$.

We know that $u2\&u1\&U > u2\&u1\&V$ and that $M(u2\&u1\&U) < M(u\&U)$. It follows that $\lceil u2,u1\&V \rceil \in \Lambda_i$, hence $\lceil u,V \rceil \in \Lambda_i$ by Lemma 9.8.

6. $\lceil typ(u,U),U \rceil \in \Lambda_{i-1}$.

By Lemma 8.7 we have $typ(u,U) \geq typ(u,V)$, hence $typ(u\&U)\&U \geq typ(u\&V)\&V$.

Now by our induction hypothesis on i it follows that $\lceil typ(u,V),V \rceil \in \Lambda_{i-1}$ and therefore $\lceil u,V \rceil \in \Lambda_i$ by clause 6.

So our proposition is proved, and it follows immediately that $u\&U \geq v\&V$, $\lceil u,U \rceil \in \Lambda_i$ imply $\lceil v,V \rceil \in \Lambda_i$. This proves our lemma.

<u>COROLLARY 10.3. Closure for Λ_i.</u>

If $i \in \mathbb{N}$, $\lceil u,U \rceil \in \Lambda_i$ and $u > v$ then $\lceil v,U \rceil \in \Lambda_i$.

As a consequence we have:

THEOREM 10.1. Closure for Λ_∞ .

If $\lceil u,U\rceil \in \Lambda_\infty$ and $u > v$ then $\lceil v,U\rceil \in \Lambda_\infty$.

11. Normability for Λ_∞.

In this section we will prove that $\lceil u,U\rceil \in \Lambda_\infty$ implies that u is normable. It then follows from Theorem 7.1 that u strongly normalizes. In order to prove that u is normable we will assign to certain sequences $U \in \Delta^*$ a sequence $s(U) \in (UG)^*$. If the assignment is not possible then we will write as before, $s(U) = \Box$.

DEFINITION 11.1. 1. $s(\emptyset) = \emptyset$.

$$2.\ s(u\&U) = \begin{cases} c_0^\alpha \& s(U) \text{ if } s(U) \neq \Box,\ fn(u,s(U)) \neq \Box \\ \qquad \text{and } fn(u,s(U))\uparrow = \alpha. \\ \Box \quad \text{otherwise} \end{cases}$$

LEMMA 11.1. If $s(U) \neq \Box$ then $L(s(U)) = L(U) = n$ and $fn(\varphi_k\langle k\rangle U,\ s(U))\uparrow = \langle k\rangle s(U)\uparrow$ for $k \leq n$.

Proof: By induction on U.

Our second lemma gives a relation between norms and typing.

LEMMA 11.2. If $U \in \Delta^*$, $s(U) \neq \Box$ and $typ(u,U) \neq \Box$ then either $fn(typ(u,U),\ s(U)) = fn(u,s(U)) = \Box$ or $fn(typ(u,U),s(U))\uparrow = fn(u,s(U))\uparrow$.

Proof: By induction on u.

We consider the case that u = [u1]u2.

Then typ(u,U) = [u1]typ(u2,u1&U) and typ(u2,u1&U) $\neq \square$.

If fn(u1,s(U)) = $\square$ then fn(typ(u,U),s(U)) = fn(u,s(U)) = $\square$.

Now assume that fn(u1,s(U)) $\neq \square$ and put fn(u1,s(U))$\uparrow$ = α. Then it follows that s(u1&U) = c_0^{α}&s(U) $\neq \square$.

If fn(typ(u2,u1&U),s(u1&U)) = $\square$ then also fn(u2,s(u1&U)) = $\square$ by the induction hypothesis, and therefore fn(typ(u,U),U) = fn(u,U) = $\square$.

So let us assume fn(typ(u2,u1&U),s(u1&U)) $\neq \square$.

Putting fn(typ(u2,u1&U),s(u1&U))$\uparrow$ = β we have by the induction hypothesis fn(u2,s(u1&U))$\uparrow$ = β and also fn(u2,g&s(U))$\uparrow$ = β for g $\in \alpha$. Hence fn(typ(u,U),s(U))$\uparrow$ = fn(u,s(U))$\uparrow$ = $\alpha \rightarrow \beta$.

<u>LEMMA 11.3</u>. If $\lceil$u,U$\rceil \in \Lambda_i$ then s(U) $\neq \square$ and fn(u,s(U)) $\neq \square$.

<u>Proof</u>: By induction on $\lceil$u,U$\rceil \in \Lambda_i$.

We consider clause 3: u = τ. We only have to show that s(U) $\neq \square$. If U = $\emptyset$ then s(U) = $\emptyset$, and if U = v&V then we have $\lceil$v,V$\rceil \in \Lambda_i$, so by the induction hypothesis s(V) $\neq \square$ and fn(v,s(V)) $\neq \square$ and therefore s(U) $\neq \square$.

We will also consider clause 4: u = <u1>u2.

We have typ(u,U) = $\square$, $\lceil$u1,U$\rceil \in \Lambda_i$, $\lceil$u2,U$\rceil \in \Lambda_i$,

typ(u1,U) $\succ$ v1 and u2 $\succ$ [v1]v2.

By the induction hypothesis fn(u1,s(U)) $\neq \square$ and fn(u2,s(U)) $\neq \square$.

Putting fn(u1,s(U))$\uparrow$ = α we have fn(typ(u1,U),s(U))$\uparrow$ = α by Lemma 11.2 and fn(v1,s(U))$\uparrow$ = α by Lemma 7.5.

Also by Lemma 7.5 fn(u2,s(U))$\uparrow$ = fn([v1]v2,s(U))$\uparrow$ = $\alpha \rightarrow \beta$ for some β,hence fn(u,s(U)) $\neq \square$.

We leave the other cases to the reader.

As a consequence we have

<u>THEOREM 11.1. Strong normalization for Λ_∞</u>

If $\lceil u,U \rceil \in \Lambda_\infty$ then u strongly normalizes.

ACKNOWLEDGEMENT

I want to express my gratitude to R. Nederpelt for his encouragement and his careful reading of the original text, where he suggested some improvements and detected a serious error.

REFERENCES

[1] de Bruijn, N.G. Lambda calculus notation with nameless dummies, a tool for automatic formula manipulation, with application to the Church-Rosser theorem. Kon. Ned. Akad. Wetensch. Proceedings Ser. A,75 (= Indag. Math. 34), 381-392 (1972).

[2] de Bruijn, N.G. A survey of the Project AUTOMATH. To H.B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism, 579-606, Academic Press (1980).

[3] van Daalen, D.T. The language theory of AUTOMATH. Thesis. Eindhoven University of Technology, Dept. of Mathematics (1980).

[4] Gandy, R.O. Proofs of Strong Normalization. To H.B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism, 457-477. Academic Press (1980).

[5] Martin-Löf, P. An intuitionistic Theory of Types. Report. University of Stockholm, Mathematical Inst. (1972).

[6] Nederpelt,R.P. Strong Normalization in a typed lambda calculus with lambda structured types.
Thesis. Eindhoven University of Technology, Dept. of Mathematics (1973).

[7] de Vrijer, R. Exactly estimating functionals and strong normalization. Preprint (1983).