COMPLETENESS OF LOGIC OF PARTIAL TERMS

VINCENZO SALIPANTE

IAM-01-005

November, 2001

COMPLETENESS OF LOGIC OF PARTIAL TERMS

VINCENZO SALIPANTE

Abstract. In this paper we prove completeness for classical first order logic of partial terms by making use of *Deduction chains* (see Schütte [5]). This technique has some advantages, as it involves no embedding and hence, in the course of the proof, the role played by partiality is clearly brought out.

ACM Subject Classification. F.4.1 (Mathematical Logic)

1. Introduction

Systems of *Explicit Mathematics* have been proposed for an axiomatic representation of mathematical practice in Bishop's approach to constructive mathematics. The logic of these systems is Beeson's *logic of partial terms*, LPT, (see Beeson [1]) in which, as pointed out by Feferman, reasoning about partial functions can be carried out *directly* (see Feferman [2]).

One of the possibilities, among several others, to prove completeness for LPT, is given by translating it into classical first order predicate logic. We can achieve this by constructing for each choice of theory T in LPT, a corresponding theory T^* in the classical predicate calculus by choosing a $n+1$-ary relation letter R_f to correspond to each n-ary function letter in LPT. In addition, the language of T^* contains the same constants and relation letters as T, but *no function letters*. We construct for each term t of T a formula $u \simeq t$ of T^*, expressing "u is defined and equal to t", where we can think of $R_f(x, y)$ as expressing that $f(x) \simeq y$. Hence we interpret each formula A of T by a formula A^* of T^* in such a way that $T \vdash A$ if and only if $T^* \vdash A^*$.

In the following we propose a direct technique (as no embedding is involved) in which partiality itself is clearly brought out.

In section 2 we introduce the syntax and semantics of LPT. In section 3 we prove the completeness of LPT by making use of *Deduction chains*.

2. The Logic of Partial Terms

2.1. **The Syntax of LPT.** In the countable first order language with equality $\mathcal{L}_1$ of LPT, all the primitive symbols are among the following:

(1) Logical symbols: $\sim$, $\vee$, $\wedge$, $\forall$ and $\exists$.

Date: November 30th, 2001.

Key words and phrases. Logic of Partial Terms, Proof Theory, Completeness, Deduction Chains.

Research supported by the NSF under grant number 11-303.

I am grateful to Prof. G. Jäger and Dr. Thomas Strahm for helpful comments and remarks on an earlier version of this paper.

(2) Parentheses and commas: (,) , [,] , .

(3) Variables: x_1, x_2, $x_3, ...$

(4) Constants: a_1, a_2, $a_3, ...$

(5) Function letters:

- f^1_1, $f^1_2, ...$ (of arity 1) ...
- f^2_1, $f^2_2, ...$ (of arity 2) ...
- ...

(6) Relation letters:

- R^1_1, $R^1_2, ...$ (of arity 1) ...
- R^2_1, $R^2_2, ...$ (of arity 2) ...
- ...

(7) The unary symbol $\downarrow$ (definedness) and the binary symbol $=$ (equality).

Our first order language $\mathcal{L}_1$ is determined by the set of primitive symbols (described above) together with definitions of the notions of *term* and *formula*.

We will define, inductively, the notion of *term* of $\mathcal{L}_1$ as follows:

(1) Variables and constants are terms.

(2) If $t_1, ..., t_n$ are terms and f^n_i is a function letter, then $f^n_i(t_1, ..., t_n)$ is a term.

(3) The terms of $\mathcal{L}_1$ are only those things generated by clauses (1) and (2).

The *positive atomic formulae* of $\mathcal{L}_1$ are expressions of the form $t\downarrow$, $(t_1 = t_2)$ and $R^n_i(t_1, ..., t_n)$ where R^n_i is a relation letter and $t_1, ..., t_n$ are terms.

The *negative atomic formulae* of $\mathcal{L}_1$ are expressions of the form $\sim A$, where A is a positive atomic formula.

The *formulae* of $\mathcal{L}_1$ are inductively defined as follows:

(1) A positive atomic formula is a formula.

(2) A negative atomic formula is a formula.

(3) If A and B are formulae, then $(A \vee B)$ is a formula.

(4) If A and B are formulae, then $(A \wedge B)$ is a formula.

(5) If A is a formula, then for any i, $\exists x_i A$ is a formula.

(6) If A is a formula, then for any i, $\forall x_i A$ is a formula.

(7) The formulae of $\mathcal{L}_1$ are only those things that are required to be so by clauses (1) - (6).

We use "A", "B",... to range over formulae.

Let x_i be a variable and suppose that $\exists x_i B$, $(\forall x_i B)$, is a formula which is a part of a formula A. Then B is called the *scope* of the particular occurrence of the quantifier $\exists x_i$, $(\forall x_i)$, in A. An occurrence of a variable x_i in A is *bound* if it falls within the scope of an occurrence of the quantifier $\exists x_i$, $(\forall x_i)$, or if it occurs inside the quantifier $\exists x_i$, $(\forall x_i)$, itself; otherwise it is *free*. A *sentence* (or *closed formula*) is a formula of $\mathcal{L}_1$ in which all the occurrences of variables are bound.

The negation* $\neg A$ of a formula A of $\mathcal{L}_1$ is inductively defined, by making use of De Morgan's laws and the law of double negation, as follows:

(1) If A is a positive atomic formula then $\neg A := \sim A$.

(2) If A is a negative atomic formula of the form $\sim B$ where B is a positive atomic formula then $\neg A := B$.

(3) If A is of the form $(B \vee C)$ then $\neg A := (\neg B \wedge \neg C)$.

(4) If A is of the form $(B \wedge C)$ then $\neg A := (\neg B \vee \neg C)$.

(5) If A is of the form $\exists x_i B$ then $\neg A := \forall x_i \neg B$.

(6) If A is of the form $\forall x_i B$ then $\neg A := \exists x_i \neg B$.

(7) The negation $\neg A$ of a formula A are only those things generated by clauses (1) - (6).

We have described our official notation; however, we shall often use an unofficial one. For example, we shall often use "x", "y", "z", ... for variables, while officially we should use x_1, x_2, x_3, ... A similar remark applies to constants, relation and function letters. We shall adopt the following abbreviations:

$$(A \to B) := (\neg A \vee B)$$

*This formulation of negation has been chosen for our Tait-style deduction system, which will be introduced in section 3.

$$(A \leftrightarrow B) := ((A \rightarrow B) \wedge (B \rightarrow A))$$

$$(s \simeq t) := (s{\downarrow} \vee t{\downarrow} \rightarrow s = t)$$

$$(s \neq t) := s{\downarrow} \wedge t{\downarrow} \wedge {\sim}(s = t)$$

Let W be a finite set $\{i_1, ..., i_n\}$ and let A_i, for every $i \in W$, be a formula. We shall adopt the following abbreviations:

$$\bigvee_{i \in W} A_i := (A_{i_1} \vee ... \vee A_{i_n})$$

and

$$\bigwedge_{i \in W} A_i := (A_{i_1} \wedge ... \wedge A_{i_n})$$

We also write $A[x/t]$ to denote the result of substituting the term t for each occurrence of the free variable x in the formula A. Similarly, $A[\vec{x}/\vec{t}]$ is the result of *simultaneously* substituting the terms $\vec{t} = t_1, ..., t_n$ for each occurrence of the free variables $\vec{x} = x_1, ..., x_n$ respectively. For substitution of terms for each occurrence of the free variables in terms we use the same notational conventions. Locally we shall adopt the following convention. In an argument, once a formula has been introduced as $A[x]$, i.e., A with a designated free variable x, we write $A(t)$ for $A[x/t]$, and similarly with more variables. We will use parentheses to avoid any possible ambiguity.

We introduce the *Hilbert-calculus* as deduction system for LPT.

Propositional Tautologies and Propositional Rules

The same as some Hilbert-calculus for classical propositional logic.

Quantifier Axioms

- $A[x/t] \wedge t{\downarrow} \rightarrow \exists x A$
- $\forall x A \wedge t{\downarrow} \rightarrow A[x/t]$

(where t may be an arbitrary term).

Quantifier Rules

$$\frac{A \rightarrow B}{\exists y A[x/y] \rightarrow B}\ (\exists)$$

(where x does not occur free in B, $y \equiv x$, or y does not occur free in A).

$$\frac{B \rightarrow A}{B \rightarrow \forall y A[x/y]}\ (\forall)$$

(where x does not occur free in B, $y \equiv x$, or y does not occur free in A).

Definedness Axioms

- $r\downarrow$, provided that r is a variable or constant.
- $f(t_1, ..., t_n)\downarrow \rightarrow t_1\downarrow \wedge ... \wedge t_n\downarrow$ for all n-ary function symbols f of L.
- $(s = t) \rightarrow s\downarrow \wedge t\downarrow$.
- $R(t_1, ..., t_n) \rightarrow t_1\downarrow \wedge ... \wedge t_n\downarrow$ for all n-ary relation symbols R of L.

Equality Axioms

- $(r = r)$, provided that r is a variable or constant of L.
- $(r = s) \rightarrow (s = r)$.
- $(s = t) \wedge (t = r) \rightarrow (s = r)$.
- $(s_1 = t_1) \wedge ... \wedge (s_n = t_n) \rightarrow f(s_1, ..., s_n) \simeq f(t_1, ..., t_n)$.
- $R(s_1, ..., s_n) \wedge (s_1 = t_1) \wedge ... \wedge (s_n = t_n) \rightarrow R(t_1, ..., t_n)$.

We denote by Th a set of sentences of LPT. Since there are countably many primitive symbols (possibly infinite), the set Th is also countable.

A $(\mathsf{Th}, \mathsf{LPT})$-*proof* is a finite sequence $A_1, ..., A_n$ of $\mathcal{L}_1$-formulae where, for each $1 \leqslant i \leqslant n$, one of the following three conditions is satisfied:

(1) A_i is an axiom of LPT,

(2) A_i is an element of Th,

(3) A_i is the conclusion of an inference rule of LPT whose premises belong to the sequence $A_1, ..., A_{i-1}$.

A *theorem* A of Th, in symbols $(\mathsf{Th}, \mathsf{LPT}) \vdash A$, is the last formula of a $(\mathsf{Th}, \mathsf{LPT})$-*proof* $A_1, ..., A_n$ (i.e. $A_n \equiv A$). In this case we say that A has a proof of length n from Th in LPT, and we denote this by $(\mathsf{Th}, \mathsf{LPT}) \vdash^n A$.

Lemma 1. *For all L terms r and s and all variables x of L we have*

$$\mathsf{LPT} \vdash r\downarrow \text{ and } \mathsf{LPT} \vdash s\downarrow \quad \Rightarrow \quad \mathsf{LPT} \vdash r[x/s]\downarrow.$$

Proof. From $\mathsf{LPT} \vdash r\downarrow$ it follows $\mathsf{LPT} \vdash \forall x(r\downarrow)$. And then

$$\mathsf{LPT} \vdash \forall x(r\downarrow) \wedge s\downarrow \rightarrow r[x/s]\downarrow$$

The statement is now proved by twice applying *Modus Ponens*. □

2.2. **The Semantics of LPT.** By a *partial structure* of the first order language $\mathcal{L}_1$ for LPT we mean a quadruple $\mathfrak{M} = (|\mathfrak{M}|, I_0, I_1, I_2)$ where $|\mathfrak{M}|$ is called the *universe* of $\mathfrak{M}$, and I_0, I_1, I_2 are functions that assign appropriate objects to the constants, function letters and relation letters of $\mathcal{L}_1$.

Specifically, a *partial* $\mathcal{L}_1$-structure is a quadruple

$$\mathfrak{M} = (|\mathfrak{M}|, I_0, I_1, I_2)$$

such that

($\mathfrak{M}$.1) The universe $|\mathfrak{M}|$ of $\mathfrak{M}$ is a non-empty set.

($\mathfrak{M}$.2) I_0 assigns to each constant c of $\mathcal{L}_1$ an element $I_0(c)$ of $|\mathfrak{M}|$ which we denote by $c^{\mathfrak{M}}$.

($\mathfrak{M}$.3) I_1 assigns to each n-place function letter f of $\mathcal{L}_1$ (with $n \geqslant 1$) an n-place *partial* function $I_1(f)$, which we denote by $f^{\mathfrak{M}}$, from $|\mathfrak{M}|^n$ to $|\mathfrak{M}|$.

($\mathfrak{M}$.4) I_2 assigns to each n-place relation letter R of $\mathcal{L}_1$ an n-place relation $I_2(R)$, which we denote by $R^{\mathfrak{M}}$, on $|\mathfrak{M}|$ (i.e. a subset of $|\mathfrak{M}|^n$).

A *valuation* γ on $|\mathfrak{M}|$ is a function which assigns to each variable x of $\mathcal{L}_1$ an element $\gamma(x)$ of $|\mathfrak{M}|$.

Let γ be a valuation on $|\mathfrak{M}|$, y a variable and j an element of $|\mathfrak{M}|$. We set, for all variables v of $\mathcal{L}_1$

$$\gamma[y:j](v) = \begin{cases} j & \text{if } v = y \\ \gamma(v) & \text{otherwise} \end{cases}$$

In defining the value $\mathfrak{M}_\gamma(t)$ of a term t of $\mathcal{L}_1$ in a partial structure $\mathfrak{M}$ with respect to a valuation γ, we have to consider the fact that the function letters can be interpreted as partial functions. A way to achieve this, is as follows. Given the universe $|\mathfrak{M}|$, we will now consider its extension $|\mathfrak{M}|_\perp$

$$|\mathfrak{M}|_\perp := \{\perp\} \cup |\mathfrak{M}|$$

where $\perp$ is an arbitrary but fixed object (depending on $|\mathfrak{M}|$) which does not belong to $|\mathfrak{M}|$.

Let $\mathfrak{M}$ be a partial $\mathcal{L}_1$-structure and γ a valuation on $|\mathfrak{M}|$. Then the value $\mathfrak{M}_\gamma(t) \in |\mathfrak{M}|_\perp$ of a term t of $\mathcal{L}_1$ is defined by induction as follows:

(1) If t is a variable, then $\mathfrak{M}_\gamma(t) := \gamma(t)$.

(2) If t is a constant, then $\mathfrak{M}_\gamma(t) := t^{\mathfrak{M}}$.

(3) Let t be a term of the form $f(t_1, ..., t_n)$ for some n-place function letter f with $n \geqslant 1$. If $\mathfrak{M}_\gamma(t_i) = \perp$, for some $1 \leqslant i \leqslant n$, or if the tuple $(\mathfrak{M}_\gamma(t_1), ..., \mathfrak{M}_\gamma(t_n))$ does not belong to the domain of $f^{\mathfrak{M}}$, then $\mathfrak{M}_\gamma(t) := \perp$; otherwise we set

$$\mathfrak{M}_\gamma(t) := f^{\mathfrak{M}}(\mathfrak{M}_\gamma(t_1), ..., \mathfrak{M}_\gamma(t_n)).$$

Let $\mathfrak{M}$ be a partial $\mathcal{L}_1$-structure and γ a valuation on $|\mathfrak{M}|$. Then the value $\mathfrak{M}_\gamma(A) \in \{\mathbf{f}, \mathbf{t}\}$ of an L formula A is defined by induction as follows:

(1) If A is the formula $t\downarrow$, then

$$\mathfrak{M}_\gamma(A) := \begin{cases} \mathbf{t} & \text{if } \mathfrak{M}_\gamma(t) \in |\mathfrak{M}| \\ \mathbf{f} & \text{if } \mathfrak{M}_\gamma(t) = \perp \end{cases}$$

(2) If A is the formula $(s = t)$, then

$$\mathfrak{M}_\gamma(A) := \begin{cases} \mathbf{t} & \text{if } \mathfrak{M}_\gamma(s) \in |\mathfrak{M}|, \mathfrak{M}_\gamma(t) \in |\mathfrak{M}| \text{ and } \mathfrak{M}_\gamma(s) = \mathfrak{M}_\gamma(t) \\ \mathbf{f} & \text{otherwise} \end{cases}$$

(3) If A is the formula $R(t_1, ..., t_n)$ for some n-place relation letter R, then

$$\mathfrak{M}_\gamma(A) := \begin{cases} \mathbf{t} & \text{if } \mathfrak{M}_\gamma(t_1) \in |\mathfrak{M}|, ..., \mathfrak{M}_\gamma(t_n) \in |\mathfrak{M}| \text{ and } (\mathfrak{M}_\gamma(t_1), ..., \mathfrak{M}_\gamma(t_n)) \in R^{\mathfrak{M}} \\ \mathbf{f} & \text{otherwise} \end{cases}$$

(4) If A is the *negtive atomic* formula $\sim B$ where B is a positive atomic formula, then

$$\mathfrak{M}_\gamma(A) := \begin{cases} \mathbf{t} & \text{if } \mathfrak{M}_\gamma(B) = \mathbf{f} \\ \mathbf{f} & \text{if } \mathfrak{M}_\gamma(B) = \mathbf{t} \end{cases}$$

(5) If A is the formula $(B \vee C)$, then

$$\mathfrak{M}_\gamma(A) := \begin{cases} \mathbf{t} & \text{if } \mathfrak{M}_\gamma(B) = \mathbf{t} \text{ or } \mathfrak{M}_\gamma(C) = \mathbf{t} \\ \mathbf{f} & \text{otherwise} \end{cases}$$

(6) If A is the formula $(B \wedge C)$, then

$$\mathfrak{M}_\gamma(A) := \begin{cases} \mathbf{t} & \text{if } \mathfrak{M}_\gamma(B) = \mathbf{t} \text{ and } \mathfrak{M}_\gamma(C) = \mathbf{t} \\ \mathbf{f} & \text{otherwise} \end{cases}$$

(7) If A is the formula $\exists x B$, then

$$\mathfrak{M}_\gamma(A) := \begin{cases} \mathbf{t} & \text{if there exists an } j \in |\mathfrak{M}| \text{ s. t. } \mathfrak{M}_{\gamma[x=j]}(B) = \mathbf{t} \\ \mathbf{f} & \text{otherwise} \end{cases}$$

(8) If A is the formula $\forall x B$, then

$$\mathfrak{M}_\gamma(A) := \begin{cases} \mathbf{t} & \text{if } \mathfrak{M}_{\gamma[x=j]}(B) = \mathbf{t}, \text{ for all } j \in |\mathfrak{M}| \\ \mathbf{f} & \text{otherwise} \end{cases}$$

By $(\mathfrak{M}, \mathsf{LPT}) \vDash A$, we mean that A is LPT-*valid* in $\mathfrak{M}$ (i.e. $\mathfrak{M}_\gamma(A) = \mathbf{t}$ for all valuations γ on $|\mathfrak{M}|$).

$\mathfrak{M}$ is called an LPT-*model* of the set Th of $\mathcal{L}_1$-sentences, $(\mathfrak{M}, \mathsf{LPT}) \vDash \mathsf{Th}$, if all sentences from Th are LPT-valid in $\mathfrak{M}$.

A is a *logical consequence* of Th, $(\mathsf{Th}, \mathsf{LPT}) \vDash A$, if A is LPT-valid in all LPT models of Th.

3. Completeness of LPT

In order to prove LPT-completeness, we need to carry out some preliminary work. We introduce an additional deduction system for LPT (other than the so-called *Hilbert calculus*), which we denote by PSC_0 (see Tait [5], [6])[†]. Rather than deriving single formulae we shall derive finite sequences of them $\Phi := A_1, A_2, ..., A_n$ meaning "A_1 or A_2 or ... or A_n". The PSC_0-deduction system has the *De Morgan* symmetries of classical logic built in and our previous treatment of negation is suitable for it.

The logical axioms of PSC_0 are

$$\Phi_0, \neg A, \Phi_1, A, \Phi_2$$

All the inference rules are among the following:

$$\frac{\Phi, A, \Psi \quad \Phi, B, \Psi}{\Phi, A \wedge B, \Psi} \ (\wedge)$$

[†]The subscript 0 denotes the new version of the $(\exists)$-inference rule.

$$\frac{\Phi, A, B, \Psi}{\Phi, A \vee B, \Psi}\ (\vee)$$

$$\frac{\exists x A, \Phi, A[x/t], \Psi \quad \exists x A, \Phi, t{\downarrow}, \Psi}{\Phi, \exists x A, \Psi}\ (\exists)$$

$$\frac{\Phi, A[x/y], \Psi}{\Phi, \forall x A, \Psi}\ (\forall)$$

where in $\forall$, y is not free in the conclusion. Provided that t is a constant or variable and is therefore defined, we are allowed to remove the second premise in $\exists$.

We write $\Vdash^n \Phi$ to mean there is a $\mathsf{PSC_0}$-proof of Φ from logical axioms with depth smaller than or equal to n.

Lemma 2. *For every sequence* Φ *and for every* n

$$\Vdash^n \Phi \quad \Rightarrow \quad \mathsf{LPT} \vdash \bigvee \Phi$$

Proof. The proof is immediate by an easy inductive argument on n.

□

Remark 3.

In the proof of Lemma 2 we only make one use of the Equality and Definedness axioms of LPT, that is

$$r{\downarrow}, \text{ provided that } r \text{ is a variable or a constant,}$$

which is required for the treatment of the $\exists$-inference rule.

We say that a sequence is *reducible* if it contains a formula which is not atomic. By the *distinguished formula* of a sequence Φ we mean the *non-atomic* formula of Φ which occurs furthest to the right.

We denote the enumeration of all terms of LPT by

$$(t_i)_{i \in \mathbb{N}} = t_0, ..., t_i, ...$$

Let Th be a set of sentences. We denote the universal closure of equality and definedness axioms by (Def + Eq). The following

$$(F_i)_{i \in \mathbb{N}} = F_0, ..., F_i, ...$$

is an enumeration of all and only sentences from $\mathsf{Th} \cup (\mathsf{Def} + \mathsf{Eq})$.

A *Deduction chain*, with respect to $\mathsf{Th} \cup (\mathsf{Def} + \mathsf{Eq})$, for a sequence Φ is a (possibly infinite) sequence of finite sequences

$$\Phi_0, \Phi_1, \Phi_2, ...$$

constructed as follows:

(1) The initial sequence Φ_0 of the D-chain is the sequence $\neg F_0, \Phi$.

(2) If a sequence Φ_n of the D-chain is an axiom of $\mathsf{PSC_0}$, then it is the last sequence of the D-chain. We say that the D-chain has length n.

(3) If a sequence Φ_n of the D-chain is not an axiom of $\mathsf{PSC_0}$ and is not reducible, then the D-chain has as immediate successor Φ_{n+1} of Φ_n, the sequence $\neg F_{n+1}, \Phi_n$.

(4) If a sequence Φ_n of the D-chain is not an axiom of $\mathsf{PSC_0}$ and it is reducible, in such a way that Φ_n has the form

$$\Psi_n, A, B_1, ..., B_n$$

where A is a non-atomic formula and $B_1, ..., B_n$ are atomic formulae (A is the distinguished formula of Φ_n) then, the D-chain has as immediate successor Φ_{n+1} of Φ_n, the sequence which is determined by $\Phi_0, ..., \Phi_n$ as follows:

(a) If A is the formula $C_0 \vee C_1$, then Φ_{n+1} is the sequence

$$\neg F_{n+1}, \Psi_n, C_0, C_1, B_1, ..., B_n$$

(b) If A is the formula $C_0 \wedge C_1$, then Φ_{n+1} is given by one of the following two sequences

$$\neg F_{n+1}, \Psi_n, C_0, B_1, ..., B_n$$

or

$$\neg F_{n+1}, \Psi_n, C_1, B_1, ..., B_n$$

(c) If A is the formula $\exists x C$, then let t_i be the first term in the enumeration of all LPT-terms

$$(t_i)_{i \in \mathbb{N}} = t_0, ..., t_i, ...$$

such that neither $C(t_i)$ nor $t_i{\downarrow}$ belongs to $\Phi_0, \Phi_1, ...\Phi_n$.

We will distinguish the following two cases:

(i) If t_i is a variable or a constant, then the immediate successor Φ_{n+1} is uniquely determined by the sequence with the following configuration

$$\neg F_{n+1}, \exists x C, \Psi_n, C(t_i), B_1, ..., B_n$$

(ii) otherwise, Φ_{n+1} is given by one of the following two sequences

$$\neg F_{n+1}, \exists xC, \Psi_n, t_i\downarrow,\ B_1, ..., B_n$$

or

$$\neg F_{n+1}, \exists xC, \Psi_n, C(t_i), B_1, ..., B_n$$

(d) If A is the formula $\forall xC$. Then Φ_{n+1} is the sequence

$$\neg F_{n+1}, \Psi_n, C(u_i), B_1, ..., B_n$$

where u_i is any new variable which does not belong to Φ_n.

D-chains are therefore formed inversely to the basic inference-rules of $\mathsf{PSC_0}$.

Principal Syntactic Lemma 4. *If every D-chain of Φ is finite, then there are finitely many elements B_1,...,B_n of* $\mathsf{Th} \cup (\mathsf{Def} + \mathsf{Eq})$ *such that*

$$\Vdash \neg B_1, ..., \neg B_n, \Phi$$

Principal Semantic Lemma 5. *Let Φ the sequence A_1,...,A_n. If there is an infinite D-chain of Φ, then there is a* partial-structure $\mathfrak{U}$ *and an valuation β on $|\mathfrak{U}|$ with*

(1) $(\mathfrak{U}, \mathsf{LPT}) \vDash \mathsf{Th} \cup (\mathsf{Def} + \mathsf{Eq})$

(2) $\mathfrak{U}_\beta(A_1 \vee ... \vee A_n) = \mathbf{f}$

Note that $\mathsf{Def} + \mathsf{Eq}$ are valid in any model.

Since we are considering the case in which Φ is not derivable, this procedure must fail to produce a derivation, and out of the failure we can construct an interpretation in which Φ is false. From these two lemmata we obtain

Completeness theorem 6. *Let Φ be the sequence A_1,...,A_n.*
If $(\mathsf{Th}, \mathsf{LPT}) \vDash (A_1 \vee ... \vee A_n)$*, then there are finitely many elements B_1,...,B_n of* $\mathsf{Th} \cup (\mathsf{Def} + \mathsf{Eq})$ *such that*

$$\Vdash \neg B_1, ..., \neg B_n, \Phi$$

Proof. Let $(\mathsf{Th}, \mathsf{LPT}) \vDash (A_1 \vee ... \vee A_n)$. By Principal Semantic Lemma, every D-chain of Φ is finite. The conclusion follows from Principal Syntactic Lemma

$$\Vdash \neg B_1, ..., \neg B_n, \Phi$$

□

Adequacy 7. *We have for all sets* Th *of $\mathcal{L}_1$-sentences and all $\mathcal{L}_1$-formulae A that*

$$(\mathsf{Th}, \mathsf{LPT}) \models A \quad \Leftrightarrow \quad (\mathsf{Th}, \mathsf{LPT}) \vdash A$$

Proof. The proof that our system is *sound* (i.e. the direction from right to left) is fairly easy by induction on the length of a proof of any given theorem. Simply note that the axioms are valid and each of the rules preserves validity.
For the converse, let $(\mathsf{Th}, \mathsf{LPT}) \models A$. Then there are finitely many elements $\mathsf{Th} \cup (\mathsf{Def} + \mathsf{Eq})$ such that

$$\Vdash \neg B_1,...,\neg B_n,A$$

From Lemma 2, we obtain

$$(\mathsf{Th}, \mathsf{LPT}) \vdash \bigwedge_{i=1}^{n} B_i \rightarrow A$$

For all $1 \leqslant i \leqslant n$, B_i is an element of $\mathsf{Th} \cup (\mathsf{Def} + \mathsf{Eq})$ and so

$$(\mathsf{Th}, \mathsf{LPT}) \vdash B_i$$

for every $1 \leqslant i \leqslant n$. Adequacy for LPT is obtained by applying *Modus Ponens* n-many times. □

Since LPT is complete, it is not difficult to see that *Compactness* and *Löwenheim-Skolem* theorems can be also proved for it.

3.0.1. *Cut-Elimination for Theories.* Generally, if we want to make $\mathsf{PSC_0}$-derivations from certain additional *non-logical* axioms NLAX, for example the *basic theory of operations and numbers*, BON, then

$$\Vdash \neg \mathsf{NLAX}, A$$

Although this latter derivation has a cut-free proof in $\mathsf{PSC_0}$, we need Cut in order to derive the formula A itself from NLAX as follows

$$\frac{\mathsf{NLAX} \quad \mathsf{NLAX} \rightarrow A}{A}$$

Thus in the presence of non-logical axioms, we cannot expect to have (full) Cut-Elimination.

3.1. **Proof of the Principal Syntactic Lemma.**

Proof. In order to prove the *Principal Syntactic Lemma* we need the following version of **König's Lemma**,

Lemma 8. *If a sequence Φ has infinitely many D-chains, then one of these chains is infinite.*

Proof. see Schütte [5]. □

Let Φ be a sequence such that every D-chain of Φ is finite. It follows from König's Lemma that there are only finitely many D-chains of Φ. Let m be the maximal length of such a D-chain. We prove the following claim by induction on $k \leqslant m$.

If the sequence Ψ occurs in the $(m-k)$-th place of a D-chain of Φ, then

- $\Vdash \Psi$, if $k = 0$
- $\Vdash \neg F_m, \Psi$, if $k = 1$
- $\Vdash \neg F_m, ..., \neg F_{m-(k-1)}, \Psi$, if $k \geqslant 2$

We consider the following cases:

(1) if Ψ is the last member of a D-chain of Φ, then Ψ is an axiom of PSC_0, and hence the above three conditions hold.

(2) if Ψ is not the last member of a D-chain of Φ, then it is $k \geqslant 1$, and it is either the conclusion of a disjunction, conjunction or quantifier rule of PSC_0 of the form

$$\frac{\Psi_{i(i=1,2 \text{ or } i=1)}}{\Psi} \ (\mathcal{S})$$

where, for every $i = 1, 2$ or $i = 1$, the sequence $\neg F_{m-(k-1)}, \Psi_i$ is the immediate successor of Ψ in the D-chain of Φ.

By I.H., it follows, for all $i = 1, 2$ or $i = 1$,

(1) $\Vdash \neg F_m, \Psi_i$, if $k = 1$

(2) $\Vdash \neg F_m, ..., \neg F_{m-(k-1)}, \Psi_i$, if $k \geqslant 2$

By applying $(\mathcal{S})$, we get

(3) $\Vdash \neg F_m, \Psi$, if $k = 1$

(4) $\Vdash \neg F_m, ..., \neg F_{m-(k-1)}, \Psi$, if $k \geqslant 2$

It follows, by taking $m = k$, that Ψ is the initial sequence of a D-chain of Φ (i.e. the sequence $\neg F_0, \Phi$). Hence

$$\Vdash \neg F_m, ..., \neg F_0, \Phi$$

where $F_m, ..., F_0$ are elements of $\mathsf{Th} \cup (\mathsf{Def} + \mathsf{Eq})$. □

3.2. **Proof of the Principal Semantic Lemma.**

Proof. Let Φ_0,Φ_1,Φ_2,... be an *infinite* D-chain of Φ (which does not contain $\mathsf{PSC_0}$-axioms). We define

$$K := \bigcup_{i \in \mathbb{N}} \mathsf{Set}(\Phi_i)$$

where $\mathsf{Set}(\Phi_i)$ is the set of all formulae belonging to Φ_i, with $i \in \mathbb{N}$.

We consider the following properties of this chain:

(1) If an atomic formula occurs in a sequence Φ_n, then it occurs in every sequence Φ_k, with $k \geqslant n$.

(2) If a non-atomic formula B occurs in a sequence Φ_n, then there is a sequence Φ_k, with $k \geqslant n$, which has B as distinguished formula.

(3) If $\exists x B$ is an element of K, then, for all terms t_i,

  (3.1) either $B(t_i)$ or $t_i{\downarrow}$ is an element of K,

  (3.2) in the case in which t_i is a variable or constant then $B(t_i)$ is an element of K.

(4) There is no atomic formula B, such that both B and $\sim B$ belong to K.

Lemma 9. *Let t be a variable or constant, then $\sim t{\downarrow}$ belongs to K.*

Proof. By definiton of D-chain, $\neg \forall y(y{\downarrow}) \in K$ and by definition of $\neg$, $\exists y(\sim y{\downarrow}) \in K$. From property (3.2), it follows

$$\sim t{\downarrow} \in K.$$

□

From K we construct a preliminary *partial counter-model*

$$\mathfrak{P} = (|\mathfrak{P}|, I_0, I_1, I_2)$$

such that

($\mathfrak{P}$.1) The universe $|\mathfrak{P}|$ of $\mathfrak{P}$ is the set $\{t_i | {\sim} t_i{\downarrow} \in K, (i \in \mathbb{N})\}$.

($\mathfrak{P}$.2) I_0 assigns to each constant c of $\mathcal{L}_1$ the constant itself, $c^{\mathfrak{P}} := c$.

($\mathfrak{P}$.3) I_1 assigns to each n-place function letter f of $\mathcal{L}_1$ (with $n \geqslant 1$) an n-place *partial* function $f^{\mathfrak{P}}$ from $|\mathfrak{P}|^n$ to $|\mathfrak{P}|$ and, for all tuples $(t_1, ..., t_n) \in |\mathfrak{P}|^n$, it is defined as

$$f^{\mathfrak{P}}(t_1, ..., t_n) := \begin{cases} f(t_1, ..., t_n) & \text{if } {\sim} f(t_1, ..., t_n){\downarrow} \in K \\ \uparrow & \text{otherwise} \end{cases}$$

($\mathfrak{P}$.4) For each n-place relation letter R of $\mathcal{L}_1$, I_2 is a function from $|\mathfrak{P}|^n$ to $\{\mathbf{f}, \mathbf{t}\}$, and, for all tuples $(t_1, ..., t_n) \in |\mathfrak{P}|^n$, it is defined as

$$R^{\mathfrak{P}}(t_1, ..., t_n) := \begin{cases} \mathbf{t} & \text{if } {\sim} R(t_1, ..., t_n) \in K \\ \mathbf{f} & \text{if } {\sim} R(t_1, ..., t_n) \notin K \end{cases}$$

By *preliminary partial counter-model* we mean that the definition ($\mathfrak{P}$.4) also comprises the case in which $R^{\mathfrak{P}}(t_1, t_2)$ is of the form $t_1 = t_2$.

The *valuation* α on $|\mathfrak{P}|$ is a function which assigns to each variable x the variable itself, $\alpha(x) := x$.

Lemma 10. *Let $t_1, ..., t_n$, be arbitrary terms. Then, for all n-place function letter f (with $n \geqslant 1$) and all n-place relation letter R, we have*

(i) $f(t_1, ..., t_n){\downarrow} \in K$ *or* ${\sim} t_1{\downarrow} \vee ... \vee {\sim} t_n{\downarrow} \in K$

(ii) $R(t_1, ..., t_n) \in K$ *or* ${\sim} t_1{\downarrow} \vee ... \vee {\sim} t_n{\downarrow} \in K$

Proof.

(i) By definition of D-chain, it follows $\neg(f(t_1, ..., t_n){\downarrow} \to t_1{\downarrow} \wedge ... \wedge t_n{\downarrow}) \in K$. Then, by definition of $\neg$, $f(t_1, ..., t_n){\downarrow} \wedge \neg(t_1{\downarrow} \wedge ... \wedge t_n{\downarrow}) \in K$ and $(f(t_1, ..., t_n){\downarrow} \wedge ({\sim} t_1{\downarrow} \vee ... \vee {\sim} t_n{\downarrow})) \in K$. By considering the premises of the ($\wedge$)-inference rule, we obtain $f(t_1, ..., t_n){\downarrow} \in K$ or $\sim t_1{\downarrow} \vee ... \vee \sim t_n{\downarrow} \in K$.[‡]

(ii) The proof goes analogously as in (i).

□

Corollary 11. *Let $t_1, ..., t_n$ be arbitrary terms. Then, for all n-place function letter f (with $n \geqslant 1$) and all n-place relation letter R, we have*

$$\sim f(t_1, ..., t_n){\downarrow} \in K \quad \Rightarrow \quad \sim t_1{\downarrow} \vee ... \vee \sim t_n{\downarrow} \in K$$

and

$$\sim R(t_1, ..., t_n) \in K \quad \Rightarrow \quad \sim t_1{\downarrow} \vee ... \vee \sim t_n{\downarrow} \in K$$

Lemma 12. *Let t be an arbitrary term. If t belongs to $|\mathfrak{P}|$ then*

(i) *$\mathfrak{P}_\alpha(t)$ is defined (i.e. $\mathfrak{P}_\alpha(t) \in |\mathfrak{P}|$)*

(ii) *$\mathfrak{P}_\alpha(t) = t$*

Proof. We use induction on the complexity of the term t. Assume that t is in $|\mathfrak{P}|$, we have

(a) t is a variable,

$$\mathfrak{P}_\alpha(t) = \alpha(t),\ \alpha(t) \text{ is in } |\mathfrak{P}| \text{ (by Lemma 9) and } \alpha(t) = t.$$

(b) t is a constant,

$$\mathfrak{P}_\alpha(t) = t^{\mathfrak{P}},\ t^{\mathfrak{P}} \text{ is in } |\mathfrak{P}| \text{ (by Lemma 9) and } t^{\mathfrak{P}} = t.$$

(c) t is of the form $f(t_1, ..., t_n)$, for a n-place function letter f (with $n \geqslant 1$) and for t_i arbitrary terms (with $1 \leqslant i \leqslant n$). Since $f(t_1, ..., t_n)$ is in $|\mathfrak{P}|$ then, it follows from Corollary 11

$$\sim f(t_1, ..., t_n){\downarrow} \in K \quad \Rightarrow \quad \sim t_1{\downarrow} \vee ... \vee \sim t_n{\downarrow} \in K$$

Hence all terms t_i, (with $1 \leqslant i \leqslant n$), are in $|\mathfrak{P}|$. This means, by I.H., that $\mathfrak{P}_\alpha(t_i)$ is in $|\mathfrak{P}|$, $\mathfrak{P}_\alpha(t_i) = t_i$ (for all $1 \leqslant i \leqslant n$), the tuple $(\mathfrak{P}_\alpha(t_1), ..., \mathfrak{P}_\alpha(t_n))$ belongs to the domain of $f^{\mathfrak{P}}$ and the value $\mathfrak{P}_\alpha(f(t_1, ..., t_n))$ of t in $\mathfrak{P}$ with respect to the valuation α belongs to $|\mathfrak{P}|$. Then,

$$\mathfrak{P}_\alpha(f(t_1, ..., t_n)) = f(t_1, ..., t_n)$$

[‡]More formally, we should consider the universal closure of definedness axioms and use variables or constants (since they are always defined) to witness the occurrence of the existential quantifier.

□

Lemma 13. *For all positive atomic formulae of the form* $R(t_1, ..., t_n)$*, if* $\sim R(t_1, ..., t_n)$ *is in* K*, then the tuple* $(\mathfrak{P}_\alpha(t_1), ..., \mathfrak{P}_\alpha(t_n))$ *belongs to the domain of* $R^{\mathfrak{P}}$*.*

Proof. Let $R(t_1, ..., t_n)$ be a positve atomic formula for an n-place relation letter R and for t_i arbitrary terms (with $1 \leqslant i \leqslant n$). If $\sim R(t_1, ..., t_n)$ is in K, then it follows from Corollary 11, that all terms t_i, (with $1 \leqslant i \leqslant n$), are in $|\mathfrak{P}|$ (or, equivalently, $\sim t_i\downarrow$ is in K, for $i = 1, ..., n$). Then, for all $1 \leqslant i \leqslant n$, $\mathfrak{P}_\alpha(t_i)$ is in $|\mathfrak{P}|$ and the tuple $(\mathfrak{P}_\alpha(t_1), ..., \mathfrak{P}_\alpha(t_n))$ belongs to the domain of $R^{\mathfrak{P}}$. □

We will show that under this partial counter-interpretation $\mathfrak{P}$ and the valuation α to each variable, every formula B occurring in K is false. So, if B is any formula, then

$$\neg B \in K \quad \Rightarrow \quad \mathfrak{P}_\alpha(B) = \mathbf{t}$$

Due to the definition of D-chains, we can prove the following lemma, by induction on the structure of the formula B occurring in K, noticing that as the sequence Φ_0,Φ_1,Φ_2,... is developed, every non-atomic formula in K will eventually "come under attention" as the first non-atomic formula at some stage.

Lemma 14. *For every formula* B*, we have*

$$B \in K \quad \Rightarrow \quad \mathfrak{P}_\alpha(B) = \mathbf{f}$$

Proof. Assume $B \in K$. The proof proceeds by induction on the complexity of B.

- $B \equiv R(t_1, ..., t_n)$ (a positive atomic formula). Then, by property (4), $R(t_1, ..., t_n) \in K$ and $\sim R(t_1, ..., t_n) \notin K$. We consider the following two cases:
  - if one of the terms $t_1, ..., t_n$, is undefined, then $\mathfrak{P}_\alpha(R(t_1, ..., t_n)) = \mathbf{f}$,
  - if $t_1, ..., t_n$, are defined, then $\mathfrak{P}_\alpha(t_i) \in |\mathfrak{P}|$, for all $1 \leqslant i \leqslant n$, and $\mathfrak{P}_\alpha(t_i) = t_i$. Hence, by definition of $R^{\mathfrak{P}}$, $\mathfrak{P}_\alpha(R(t_1, ..., t_n)) = \mathbf{f}$.
- $B \equiv \sim R(t_1, ..., t_n)$ (a negative atomic formula). Then $\sim R(t_1, ..., t_n) \in K$ and $R(t_1, ..., t_n) \notin K$. By Corollary 11, $\sim t_1\downarrow \vee ... \vee \sim t_n\downarrow \in K$. And then, by Lemma 13, the tuple $(\mathfrak{P}_\alpha(t_1), ..., \mathfrak{P}_\alpha(t_n))$ belongs to the domain of $R^{\mathfrak{P}}$. Hence $\mathfrak{P}_\alpha(\sim R(t_1, ..., t_n)) = \mathbf{f}$ and $\mathfrak{P}_\alpha(R(t_1, ..., t_n)) = \mathbf{t}$.

- $B \equiv C_0 \wedge C_1$. $B \in K$, then there exists a k such that $B \in \Phi_k$ and B is the rightmost non-atomic formula in Φ_k or, equivalently, the distinguished formula in Φ_k. Hence, by (4.b), either $C_0 \in \Phi_{k+1}$ or $C_1 \in \Phi_{k+1}$ and then, by definition of K, either $C_0 \in K$ or $C_1 \in K$.

  By I.H. $\mathfrak{P}_\alpha(C_0) = \mathbf{f}$ or $\mathfrak{P}_\alpha(C_1) = \mathbf{f}$ and then
  $\mathfrak{P}_\alpha(C_0 \wedge C_1) = \mathbf{f}$.

- $B \equiv C_0 \vee C_1$. $B \in K$, then there is a k such that $B \in \Phi_k$ and B is the distinguished formula in Φ_k. Hence, by (4.a), both C_0 and $C_1 \in \Phi_{k+1}$ and then, by definition of K, both C_0 and $C_1 \in K$.

  By I.H. $\mathfrak{P}_\alpha(C_0) = \mathbf{f}$ and $\mathfrak{P}_\alpha(C_1) = \mathbf{f}$ and then
  $\mathfrak{P}_\alpha(C_0 \vee C_1) = \mathbf{f}$.

- $B \equiv \exists x C$. $B \in K$, then by property (3), for all terms t_i either $C(t_i)$ or $t_i{\downarrow} \in K$.

  By I.H. $\mathfrak{P}_\alpha(C(t_i)) = \mathbf{f}$ or $\mathfrak{P}_\alpha(t_i{\downarrow}) = \mathbf{f}$ and then
  $\mathfrak{P}_\alpha(\exists x C) = \mathbf{f}$.

  (Observe that if $t_i \in |\mathfrak{P}|$, then $\mathfrak{P}_\alpha(t_i{\downarrow}) = \mathbf{t}$).

- $B \equiv \forall x C$. $B \in K$, then there exists a k such that $B \in \Phi_k$ and B is the distinguished formula in Φ_k. By (4.d), $C(u_i) \in \Phi_{k+1}$ and then, by definition of K, $C(u_i) \in K$.

  By I.H. $\mathfrak{P}_\alpha(C(u_i)) = \mathbf{f}$ and then
  $\mathfrak{P}_\alpha(\forall x C) = \mathbf{f}$.

  (Observe that $u_i \in |\mathfrak{P}|$).

□

Since $\{A_1, ..., A_n\} = \mathsf{Set}(\Phi) \subset K$, it follows that $\mathfrak{P}_\alpha(A_i) = \mathbf{f}$, for all $1 \leq i \leq n$. This implies

$$\mathfrak{P}_\alpha(A_1 \vee ... \vee A_n) = \mathbf{f}.$$

Let $F_i \in \mathsf{Th} \cup (\mathsf{Def} + \mathsf{Eq})$, $(i \in \mathbb{N})$, then, by definition of D-chain, $\neg F_i \in K$. Hence

$$\mathfrak{P}_\alpha(\neg F_i) = \mathbf{f} \quad \text{and} \quad \mathfrak{P}_\alpha(F_i) = \mathbf{t}$$

(Note that $\mathsf{Th} \cup (\mathsf{Def} + \mathsf{Eq})$ is a set of sentences. Every F_i in $\mathsf{Th} \cup (\mathsf{Def} + \mathsf{Eq})$ is a *closed formula* and does not contain free variables. Hence the value of F_i in $\mathfrak{P}$ does not depend on the valuation α).
For all $1 \leq i \leq n$, F_i is an element of $\mathsf{Th} \cup (\mathsf{Def} + \mathsf{Eq})$, and then $(\mathfrak{P}, \mathsf{LPT}) \vDash F_i$, for

every $1 \leq i \leq n$. Hence

$$(\mathfrak{P}, \mathsf{LPT}) \vDash \mathsf{Th} \cup (\mathsf{Def} + \mathsf{Eq})$$

Although equality axioms are LPT-valid in $\mathfrak{P}$, it is not still guaranteed that equality symbol $=$ can be interpreted in $\mathfrak{P}$ as identity on the elements of $\mathfrak{P}$.
A way to deal with an adequate interpretation of it is as follows. For all terms s, t in $|\mathfrak{P}|$, let $\doteq$ be a new symbol defined as

$$s \doteq t :\Leftrightarrow \mathfrak{P}_\alpha(s = t) = \mathbf{t}$$

note also that,

$$s \doteq t :\Leftrightarrow \sim(s = t) \in K$$

Since Equality axioms are LPT-valid in $\mathfrak{P}$, it is easily shown that $\doteq$ is an equivalence relation on $|\mathfrak{P}|$.

We now consider the equivalence class $[t]$ of t modulo $\doteq$, such that, for all terms t in $|\mathfrak{P}|$,

$$[t] = \{s \in |\mathfrak{P}| : s \doteq t\}$$

Based on this equivalence relation we finally define our partial counter-model $\mathfrak{U}$ as follows:

$$\mathfrak{U} = (|\mathfrak{U}|, E_0, E_1, E_2)$$

($\mathfrak{U}$.1) The universe $|\mathfrak{U}|$ of $\mathfrak{U}$ is the set $\{[t] : t \in |\mathfrak{P}|\}$.

($\mathfrak{U}$.2) E_0 assigns to each constant c of $\mathcal{L}_1$ the equivalence class of c, $c^{\mathfrak{U}} := [c]$.

($\mathfrak{U}$.3) E_1 assigns to each n-place function letter f of $\mathcal{L}_1$ (with $n \geqslant 1$) an n-place *partial* function $f^{\mathfrak{U}}$ from $|\mathfrak{U}|^n$ to $|\mathfrak{U}|$ and, for all tuples $(t_1, ..., t_n)$ of $|\mathfrak{P}|^n$, it is defined as

$$f^{\mathfrak{U}}([t_1], ..., [t_n]) := \begin{cases} [f(t_1, ..., t_n)] & \text{if } f(t_1, ..., t_n) \in |\mathfrak{P}| \\ \uparrow & \text{otherwise} \end{cases}$$

($\mathfrak{U}$.4) For each n-place relation letter R of $\mathcal{L}_1$, E_2 is a function from $|\mathfrak{U}|^n$ to $\{\mathbf{f}, \mathbf{t}\}$, and, for all tuples $(t_1, ..., t_n)$ of $|\mathfrak{P}|^n$, it is defined as

$$R^{\mathfrak{U}}([t_1], ..., [t_n]) := R^{\mathfrak{P}}(t_1, ..., t_n)$$

The *valuation* β on $\mathfrak{U}$ is a function which assigns to each variable x the equivalence class of x, $\beta(x) := [x]$.

We have, for all terms t

$$t \in |\mathfrak{P}| \quad \Rightarrow \quad \mathfrak{U}_\beta(t) \in |\mathfrak{U}|.$$

We must show that $f^{\mathfrak{U}}$ and $R^{\mathfrak{U}}$ (the right-hand sides of our previous definitions) are well-defined; i.e. these definitions do not depend on the particular choice of members of the equivalence classes used.
For this, suppose that $[t_i] = [w_i]$, for $i = 1, ..., n$. Then $(\mathfrak{P}, \mathsf{LPT}) \vDash t_i = w_i$. So, by equality axioms,

$$(\mathfrak{P}, \mathsf{LPT}) \vDash f(t_1, ..., t_n) \simeq f(w_1, ..., w_n)$$

and

$$(\mathfrak{P}, \mathsf{LPT}) \vDash R(t_1, ..., t_n) \leftrightarrow R(w_1, ..., w_n).$$

Hence, by assuming that $f(t_1, ..., t_n) \in |\mathfrak{P}|$,

$$[f(t_1, ..., t_n)] = [f(w_1, ..., w_n)]$$

and

$$R^{\mathfrak{P}}(t_1, ..., t_n) \text{ iff } R^{\mathfrak{P}}(w_1, ..., w_n).$$

This is just what we wanted to prove.

Finally,

$$\mathfrak{U}_\beta(B) = \mathbf{t} \quad \Leftrightarrow \quad \mathfrak{P}_\alpha(B) = \mathbf{t}$$

Proof. By induction on the complexity of the formula B. □

Hence we have, for all terms a, b in $|\mathfrak{U}|$,

$$\mathfrak{U}_\beta(a = b) = \mathbf{t} \quad \Leftrightarrow \quad a = b$$

This means that the equality symbol $=$ is now interpreted in $\mathfrak{U}$ as *identity* on the elements of $\mathfrak{U}$. □

References

[1] Beeson, M. J., *Foundations of Constructive Mathematics*, Ergebnisse der Mathematik und ihrer Grenzgebiete; 3. Folge, Band 6. Springer-Verlag, Berlin, 1985.
[2] Feferman, S., *Definedness*, Lecture for mini-conference on Partial Functions and Programming: Foundational Questions, U.C. Irvine, May 17th, 1995.
[3] Feferman, S., Jäger, G., Strahm, T., *Explicit Mathematics*, draft version, March 6th, 2001.
[4] Shoenfield, J. R., *Mathematical Logic*, Addison-Wesley, Reading, Massachussets, 1967.
[5] Schütte, K., *Proof Theory*, Springer-Verlag, Berlin, 1977.
[6] Tait, W. W., *Normal Derivability in Classical Logic. The Syntax and Semantics of Infinitary Languages*, Lecture Notes in Mathematics, 72, Springer-Verlag, 1968.
[7] Tait, W. W., *Applications of the cut-elimination theorem to some subsystems of classical analysis*, in *Intuitionism and Proof Theory*, North Holland, Amsterdam , 1970.
[8] Wainer, S. S., *Basic Proof Theory with Applications to Computation*, Department of Pure Mathematics, Preprint Series No. 18, 1996.

Institut für Informatik und angewandte Mathematik
Universität Bern
Neubrückstrasse 10
CH-3012 Bern, Switzerland
E-mail address: salipant@iam.unibe.ch