DIPLOMARBEIT

Titel der Diplomarbeit

"Syntactical Consistency Proofs for Term Induction Revisited: Two Different Methods"

Verfasser

Michael Toppel

angestrebter akademischer Grad

Magister der Naturwissenschaften (Mag.rer.nat)

Wien, im Juli 2011

Studienkennzahl lt. Studienblatt:	A 405
Studienrichtung lt. Studienblatt:	Mathematik
Betreuer:	Dr. Jakob Kellner
	Ao.Univ.Prof. Dr. Matthias Baaz

Contents

Introduction

At the beginning of the 19^{th} century David Hilbert became interested in the idea that all of mathematics can be covered by elementary operations on finite strings of symbols. So one can speak about mathematics without referring to the "meaning" (whatever that means) of the symbols that are used. Motivated by this idea he started two programs: One to show that the extension of mathematics to the infinite does not change finite mathematics and (as a successor of the first) another one to show with elementary (finite) methods the consistency of mathematical theories. These two programmes are connected by the fact that a solution of the second leads to a solution of relevant parts of the first (see [14]). Since the syntax of ordinary first order logic amounts to finite operations on finite strings of symbols, the notions of "proof", "provability" and "consistency" are part of "finite mathematics". This was the beginning of proof theory. Of course "finite mathematics" is not a well defined concept. Several logicians have argued that it should be peano arithmetic **PA**, others have favoured primitive recursive arithmetic **PRA** (including all primitive recursive functions plus open induction).
By the work of Kurt Gödel, Hilbert's program ran aground. With his second incompleteness theorem (published 1931) Gödel showed that even for weak theories like **PA** there cannot be a consistency proof using just primitive recursive methods. Otherwise **PA** would be strong enough to prove its own consistency, contradicting the second incompleteness theorem.
In the year 1936 Gerhard Gentzen published a syntactical consistency proof of **PA** [5] and 1938 a second version [6]. His result does not contradict Gödel's incompleteness theorem, because the proof uses an induction "longer than the natural numbers" (over the well-ordering of an ordinal notation system up to ε_0). So the second incompleteness theorem implies that this ε_0-induction-principle can not be proven in **PA**. Paul Lorenzen, Kurt Schütte and others used the work of Gentzen to develop longer ordinal notation systems to show (or from another point of view, to measure) the consistency of stronger theories. In their work, they use infinite languages and calculi but still keep primitive recursivness in some sense. This part of proof theory is called ordinal analysis and can be seen as an extended Hilbert program.
It became common to speak about "the ordinal" of a given theory when talking about the consistency strength (in a primitive recursive way). This is somewhat misleading as Georg Kreisel pointed out. He used several counter examples to show that the proof theoretical ordinal (as a primitive recursive ordering) of a theory is not a robust notion. For example, an easy argument shows that the shortest primitive recursive well-ordering one needs to prove the consistency of a consistent recursive first order theory has always order-type ω. So the proof theoretical ordinal only makes sense in relation to a given notation system (if one wants to keep primitive recursiveness).
A way to deal with this problem is presented by Wolfram Pohlers in his

book [8]. Pohlers extends the language of first order logic by second order variables (but not second order quantification) to define the Π^1_1-ordinal of a theory. This notion of a proof theoretical ordinal is in fact a real ordinal (not an element of an ordinal notation system), so it is not given in a primitive recursive way.
Ordinal analysis leads to some useful applications in proof theory beyond assigning a proof theoretical strength. For several examples see [9].
In my diploma thesis I will take a closer look at the method Gentzen used in his consistency proof for **PA**. Sometimes it is supposed in the literature that Gentzen's method is essential the same as Schütte's (for example see [8, p.123]). This is somewhat misleading because Gentzen's method relies essentially on the induction schema

$$\varphi(0) \land \forall x[\varphi(x) \to \varphi(x+1)] \to \forall x \varphi(x)$$

but not of the other axioms of **PA**, except from the fact that they are all universal sentences. In contrast, Schütte's and Tait's methods seem to require other properties of the underlying theories: In the case of arithmetic all functions and predicate constants have to have a standard interpretation, and all axioms have to be true (in the standard interpretation). (In the case one wants to work in **PRA** it is useful to have only primitive recursive interpretations.) However, in this thesis we only present Gentzen's mothod and (a variant of) Schütte's, the analysis of the differences would go beyond the scope of this diploma thesis.
In Chapter 1 I will introduce an ordinal notation system up to Γ_0, and several different notions of "proof theoretical ordinal of a theory". In Chapter 2 I will give a slight generalisation of Gentzen's result which covers all primitive recursive theories T_f which include just universal sentences together with an induction schema

$$\varphi(c_1) \land ... \land \varphi(c_m) \land \forall \vec{x}[\varphi(x_1) \land ... \land \varphi(x_n) \to \varphi(f(x_1, ..., x_n))] \to \forall x \varphi(x).$$

Here one has to assume that the theory proofs in a very simple way that any of its closed terms is equal to a term build up from f and $c_1, ..., c_m$ and that all axioms except from induction are universal sentences. This will lead to the fact that every Σ^0_1-sentence which is provable in T_f is also provable in T_f without the use of the induction schema. As in Gentzen's original proof we have to assign to each deduction a rank, which is an ordinal term (in an ordinal notation system up to ε_0).
In Gentzen's proof ε_0 appears from nowhere: the way how the ranks are assigned to the deductions is rather artificial and depends on the fact that ε_0 is the right choice. (The ordinal term ε_0 is the smallest term of the notation system which can be used, because transfinite induction is provable in **PA** for every smaller term (see [6]).)
Contrary to Gentzen's method the method W. W. Tait developed in [15] has

a stronger relation to the ordinals: One uses infinite propositional logic and infinite deductions trees and every deduction in the considered theory such as **PA** can be transformed into such a tree. Since the method uses infinite deduction trees, the rank can be defined very natural as the smallest ordinal which is bigger than the ranks of all subdeductions. Tait proves in [15] how the rank raise after cut-elimination is related to the χ-function: A deduction of rank α, where the "cut-rank" is smaller or equal than $\beta + \omega^\gamma$, can be transformed in a deduction of rank $\chi(\gamma, \alpha)$ and cut-rank β. The cut-rank is defined as the smallest ordinal term (or in some approaches: ordinal) which is bigger than the complexity of any cut-formula in the deduction, so a cut-rank of 0 means cut-freeness. Therefore by repeating the procedure the rank increases and the cut-rank degreases in each step, so we end up in a cut-free deduction which rank is smaller than the smallest fix-point of χ bigger than α.
This gives a consistency proof, because cut-freeness implies the subformula property (every provable formula has an axiom as subformula). This proof uses transfinite induction up to the fix-point the Elimination Theorem gives for the embedded deductions of the considered theory. So in the case of **PA** the theorem leads to $\chi(1,0)$ which is ε_0. So in Tait's method one has only to find an embedding (i.e., a transformation which transforms ordinary (finite) deductions of the theory in infinite deductions trees) and then the proof theoretical ordinal can be calculated whereas in Gentzen's method one has to guess the proof theoretical ordinal to be able to assign ranks to deductions in a suitable way.
I will only present a variant of Tait's method, to measure the Π^1_1-ordinal, in Chapter 3. There I will consider Theories TA_f which include all definable axioms for primitive recursive functions and the induction axioms

$$\varphi(c_1) \wedge ... \wedge \varphi(c_m) \wedge \forall\vec{x}[\varphi(x_1) \wedge ... \wedge \varphi(x_n) \to \varphi(f(x_1, ..., x_n))] \to \forall x \varphi(x)$$

for one primitive recursive function f. Because I restrict myself to finding an upper bound for the Π^1_1-ordinal of TA_f, I will consider languages which contain second order variables (without second order quantification). We call the sentences of this language (pseudo) Π^1_1-sentences. This will chance the focus from consistency to the approach Wolfram Pohlers give in his book [8]. I do this because giving all the details of coding the methods primitive recursively would go beyond the extend of this diploma thesis.
In the end of this introduction I want to thank a couple of people who helped me in my academic life: Firstly my parents who gave me the confidence a child with dyslexia needs to grow up in the school system of Austria. Secondly my supervisor Jakob Kellner who took the burden to supervise a diploma thesis which has nothing to do with his personal interests. And last but of course not least Matthias Baaz for his friendly support and the supply of his proof theoretical expertise given to a student who does not even study at his university.

1 Finite dealings with infinite Ordinal Numbers

1.1 The Class of Ordinal Numbers

Two methods of syntactical consistency proofs, one invented by Gentzen and the other by Tait, will be introduced. For both methods, we assign to any deduction (as a rooted tree) a number called the rank of the deduction. But it is not sufficient to use finite numbers for the rank of a deduction. So in this chapter a notation system will be introduced which extends the natural numbers. This will be the system of ordinal notations up to the ordinal Γ_0. This presentation is closely related to [7], [8] and [12].

1.1.1 ZFC

For those readers who like to work in **ZFC** we show:

1. How one can obtain the ordinal notation system that we will use from set theory.
2. How the notations are related to the usual way of presenting ordinals in **ZFC**.
3. How one can see that the notation system is well-founded.

Note that all these questions can be deal with in many other systems that can talk about ordinals as well, so in some sense what we do with ordinals is independent from **ZFC**.
The language of set theory is the first order language with the signature $\{\in, =\}$, denoted by $\mathcal{L}(\in, =)$ (see Section 2.1). In this language it is possible to define union ($\cup$), successor ($S(x) = x \cup \{x\}$), the empty set ($\emptyset$) and the subset relation ($\subset$). To denote first order provability we will use "$\vdash$".

Definition 1.1. ZFC
The axioms of ZFC will be the following:

1. Set Existence:
$$\exists x(x = x)$$
2. Extensionality:
$$\forall x \forall y(\forall z(z \in x \leftrightarrow z \in y) \to x = y)$$
3. Foundation:
$$\forall x[\exists y(y \in x) \to \exists y(y \in x \wedge \neg \exists z(z \in x \wedge z \in y))]$$
4. Pairing:
$$\forall x \forall y \exists z(x \in z \wedge y \in z)$$

5. Union:
$$\forall x \exists y \forall z \forall w (w \in z \wedge z \in x \to w \in y)$$

6. Infinity:
$$\exists x (\emptyset \in x \wedge \forall y \in x (S(y) \in x))$$

7. Power set:
$$\forall x \exists y \forall z (z \subset x \to z \in y)$$

8. Axiom of Choice:
$$\forall x [(\forall y \in x (y \neq \emptyset)) \to (\exists f \forall y \in x f(y) \in y)]$$

together with the two following schemas:

1. For all formulas φ where x, z are not bound we have the following Comprehension axiom:
$$\forall z \exists y \forall x (x \in y \leftrightarrow x \in z \wedge \varphi).$$
We denote the set y by $\{x \in z | \varphi\}$.

2. For all formulas φ where x, y, z are not bounded we have the following Replacement axiom:
$$\forall x (\forall y \in x \exists ! z \varphi \to \exists w \forall y \in x \exists z \in w \varphi).$$

We can also define the ordered pair $\langle x, y\rangle$ and the cartesian product $x \times y$. A (two place) relation on x, as a set, is a subset of $x \times x$. We usually use the symbol R to denote relations. Instead of $\langle a, b\rangle \in R$ we also write aRb. It should be clear how to define functions $f : x \to y$, rang ($rng(f)$), domain ($dom(f)$) and the restriction $f|_z$ on a subset $z \subset x$. We define isomorphism of sets together with a relation in the usual way,
$$f : \langle A, R\rangle \to \langle B, S\rangle$$
is an isomorphism iff $f : A \to B$ is bijective and
$$\forall x \forall y (xRy \leftrightarrow f(x)Sf(y)).$$
And we write $\langle A, R\rangle \cong \langle B, S\rangle$ if there is such f.

Definition 1.2. The set $\langle A, R\rangle$ is a *total order* iff R satisfies the following:

1. $\forall x \in A \neg(xRx)$
2. $\forall x, y, z \in A(xRy \wedge yRz \to xRz)$
3. $\forall x, y \in A(x = y \vee xRy \vee yRx)$

The next definition will give us a notation for a special kind of ordering.

Definition 1.3. $\langle A, R\rangle$ is a *well-order* iff $\langle A, R\rangle$ is a total order and

$$\forall x \subset A(x \neq \emptyset \to \exists y \in A(\neg\exists z \in A(zRy))),$$

i.e., each non-empty subset of A has a R-minimal element.

The next proposition should be obvious. It shows the closed relation between well-ordering and induction over a relation.

Proposition 1.4. *$\langle A, R\rangle$ is a well-order iff*
$\forall x(\forall y \in A(\forall z \in A(zRy \to z \in x)) \to y \in x) \to \forall y(y \in A \to y \in x))$.

Remark 1.5. The formula in the only-if position at the proposition above is called the formula of transfinite induction over the well-order $\langle A, R\rangle$.

Definition 1.6. A set A is *transitive* iff $\forall x(x \in A \to x \subset A)$.

Now we are prepared to give the definition of ordinal numbers. For that, we need a special kind of a relation, called the membership relation $\in_x$ of a set x, defined by $\in_x := \{\langle y, z\rangle \in x \times x | y \in z\}$.

Definition 1.7. A set x is an *ordinal number* iff x is transitive and $\langle x, \in_x\rangle$ is a well order.

Note the class of ordinal numbers is a proper class, so being in the class of ordinal numbers, informally $x \in \mathbf{ON}$ or $\mathbf{ON}(x)$, is an abbreviation for the formula in the definition of ordinal number. This way we can introduce the notation of **classes** as some kind of metasets, so we can think of $\mathbf{ON}$ as the class defined from the formula in Definition 1.7.

Theorem 1.8. ***Ordinal Properties***

- *If $x \in \mathbf{ON}$ and $y \in x$, then $y \in \mathbf{ON}$ and $y \subset x$.*
- *If $x, y \in \mathbf{ON}$ and $\langle x, \in_x\rangle \cong \langle y, \in_y\rangle$, then $x = y$.*
- *If $x, y \in \mathbf{ON}$, then exactly one of the three statements holds: $x = y$, $x \in y$ or $y \in x$.*
- *If $x, y, z \in \mathbf{ON}$, $x \in y$ and $y \in z$, then $x \in z$.*
- *If $x \subset \mathbf{ON}$ and $x \neq \emptyset$, then x have a $\in$-least element.*
- *If (in the metalevel) $\boldsymbol{C}$ is a class, then holds:*
  If $\boldsymbol{C} \subset \mathbf{ON}$ and $\boldsymbol{C} \neq \emptyset$, then $\boldsymbol{C}$ have a $\in$-least element.

Proof. See [7, I, §7]. □

So in some meta sense **ON** is a class well-order. This makes transfinite induction possible. We will define this generally for class relations.

Definition 1.9. A class relation $\mathbf{R}$ is well-founded on a class $\mathbf{A}$ iff $\forall x \subset \mathbf{A}(x \neq \emptyset \rightarrow \exists y \in \mathbf{A}(\neg \exists z \in \mathbf{A}(z\mathbf{R}y)))$.

Of course the class notation here is an abbreviation for formulas which define the class.

To better reflect the intuition about the order of ordinals, the symbol $<$ will be used instead of $\in$ when we talk about ordinals, and we will use small greek letters to denote them. From the definition of the successor it should be clear that if $x \in \mathbf{ON}$, then also $S(x) \in \mathbf{ON}$ and the statements $\alpha < S(\alpha)$ and $\forall \beta(\beta < S(\alpha) \leftrightarrow \beta \leq \alpha)$ hold. From that it is easy to see that $\emptyset$ is the smallest ordinal and that the first ordinals are its successors. So we define: $0 := \emptyset, 1 := S(0), 2 := S(1)$ a.s.o. The next definition will give two different kinds of ordinals.

Definition 1.10. α is a *successor ordinal* iff $\exists \beta(\alpha = S(\beta))$. α is a *limit ordinal* iff $\alpha \neq 0$ and α is not a successor ordinal.

Definition 1.11. α is a *natural number* iff $\forall \beta \leq \alpha(\beta = 0 \vee \beta$ is a successor ordinal).

The next definition gives the smallest limit ordinal. Its existence is assured by Axiom 6.

Definition 1.12. ω is the set of all natural numbers.

Now we want to introduced the concept of cardinality. A set A is said to be well-orderable iff there is a well-order on A. The second axiom schema ensures that the following definition makes sense [7].

Definition 1.13. Assume a set A can be well-ordered. The function $|A| = \alpha$ iff α is the least ordinals such that there is a bijection between A and α.
$\alpha \in \mathbf{ON}$ is a *cardinal* iff $|\alpha| = \alpha$.

We used the well known aleph notation to denoting cardinals, i.e.
$\aleph_0, \aleph_1, \aleph_2, ..., \aleph_\omega, ...$ a.s.o transfinite. We say a set A is *countable* iff $|A| < \aleph_1$.

1.1.2 Ordinal Arithmetic and Definable Functions

Using the theorem of transfinite recursion (see [7, I, §9]) we can define class functions (or meta functions) by transfinite recursion on **ON**.

Definition 1.14. Addition

- $\alpha + 0 = \alpha$
- $\alpha + 1 = S(\alpha)$

- $\alpha + S(\beta) = S(\alpha + \beta)$
- If β is a limit, then $\alpha + \beta = \bigcup\{\alpha + \gamma | \gamma < \beta\}$

Definition 1.15. Multiplication

- $\alpha \cdot 0 = 0$
- $\alpha \cdot 1 = \alpha$
- $\alpha \cdot S(\beta) = \alpha \cdot \beta + \alpha$
- If β is a limit, then $\alpha \cdot \beta = \bigcup\{\alpha \cdot \gamma | \gamma < \beta\}$

Definition 1.16. Exponentiation

- $\alpha^0 = 1$
- $\alpha^{\beta+1} = \alpha^\beta \cdot \alpha$
- If β is a limit, then $\alpha^\beta = \bigcup\{\alpha^\gamma | \gamma < \beta\}$

The next proposition shows that the functions 2^α and ω^α have the same fix-points.

Proposition 1.17. *Assume $\alpha \neq \omega$ is an ordinal. Then $2^\alpha = \alpha$ iff $\omega^\alpha = \alpha$.*

Proof. See [1, III, §15]. □

For the next class function we need a well known fact about the normal form of ordinal numbers which was discovered by Cantor.

Theorem 1.18. *Cantors Normalform Theorem*
Every ordinal $\alpha > 0$ can be represented uniquely in the form

$$\alpha = \omega^{\beta_1} + ... + \omega^{\beta_n},$$

where $\omega > n > 0$ and $\alpha \geq \beta_1 \geq ... \geq \beta_n$.

Proof. See [11, IV, §14]. □

Definition 1.19. Natural Sum

- $\alpha \sharp 0 = \alpha$
- $0 \sharp \beta = \beta$
- If $\alpha = \omega^{\gamma_1} + ... + \omega^{\gamma_n}$ and $\beta = \omega^{\delta_1} + ... + \omega^{\delta_m}$, then $\alpha \sharp \beta = \omega^{\lambda_1} + ... + \omega^{\lambda_{n+m}}$ where $\lambda_i \in \{\gamma_1, ..., \gamma_n, \delta_1, ..., \delta_m\}$ and $\lambda_1 \geq ... \geq \lambda_{n+m}$.

The next two class functions are the class version of the well known order-type function and the enumeration function of a well-ordered set.

Definition 1.20. Let $\mathbf{A}$ be a class, $\mathbf{R}$ be a well-ordered class relation on $\mathbf{A}$ and $x \in \mathbf{A}$. Then $\mathrm{otyp}_{\mathbf{A}} : \mathbf{A} \to \mathbf{ON}$ is defined as follows:

- If x is the $\mathbf{R}$-smallest element in $\mathbf{A}$, then $\mathrm{otyp}_{\mathbf{A}}(x) := 0$.
- Otherwise $\mathrm{otyp}_{\mathbf{A}}(x) := \bigcup\{\mathrm{otyp}_{\mathbf{A}}(y) + 1 | y\mathbf{R}x\}$.

The order-type of $\mathbf{R}$ itself is

$$\mathrm{otyp}(\mathbf{A}) := \bigcup\{\mathrm{otyp}_{\mathbf{A}}(x) + 1 | x \in \mathbf{A}\}.$$

It should be clear that $\mathrm{otyp}(\mathbf{A})$ is either an ordinal or the whole class $\mathbf{ON}$ (see [8, Ch.3, p.26]) for every well-ordered class relation $\mathbf{R}$ over a class $\mathbf{A}$. In [8, Ch.3, p.26] is also proved that the function $\mathrm{otyp}_{\mathbf{A}}$ is a order preserving bijection, so there is an order preserving bijection

$$en_{\mathbf{A}} := \mathrm{otyp}_{\mathbf{A}}^{-1}$$

which is just the enumeration of the elements in $\mathbf{A}$.

1.1.3 Regular Ordinals, Cardinals and Club

An important concept in set theory are regular ordinals and the club filter which will be introduced in this section. Note that the class notations of subset and intersection are abbreviations for formulas.

Definition 1.21. We call a class $\mathbf{A} \subset \mathbf{ON}$ *unbounded in* $\alpha \in ON$ iff for all $\beta < \alpha$ there is a $\gamma \in \mathbf{A} \cap \alpha$ such that $\beta < \gamma$.

Definition 1.22. Assume $\alpha, \beta \in \mathbf{ON}$. A function $f : \alpha \to \beta$ is called *cofinal* iff $rng(f)$ is unbounded in β.

Definition 1.23. The *cofinality* $cf(\alpha)$ of an ordinal α is the least β such that there is a cofinal function $f : \beta \to \alpha$.

Definition 1.24. An ordinal α is *regular* iff $cf(\alpha) = \alpha$.

From that we can give the definitions of club.

Definition 1.25. Assume α is regular. A class $\mathbf{A} \subset \mathbf{ON}$ is *closed in* α iff for all $U \subset \mathbf{A} \cap \alpha$ with $U \neq \emptyset$ and $|U| < \alpha$ holds that $\bigcup U \in \mathbf{A}$.

Definition 1.26. Assume α is regular. A class $\mathbf{A}$ is *α-club* iff $\mathbf{A}$ is closed and unbounded in α.

Definition 1.27. Assume α is regular and $f : \mathbf{ON} \to \mathbf{ON}$ is order preserving. f is *α-continuous* iff $dom(f)$ is closed in α and for all $U \subset dom(f) \cap \alpha$ with $U \neq \emptyset$ and $|U| < \alpha$ holds that $\bigcup f(U) = f(\bigcup U)$.
A function f is *α-normal* iff f is α-continuous and $\alpha \subset dom(f)$.

The next theorem will ensure that the Veblen functions, defined in the next section, are normal for a adequate regular ordinal.

Theorem 1.28. *Assume α is regular.*
A class $\boldsymbol{A} \subset \boldsymbol{ON}$ is α-club iff $en_{\boldsymbol{A}}$ is α-normal.

Proof. See [8, Ch.3, p.27]. $\square$

1.1.4 Veblen Hierarchy

The Veblen Hierarchy gives most of the ordinals used in proof theory. The basis of the hierarchy is the set of *principal ordinals*

$$\mathbb{H} := \{\alpha \in \mathbf{ON} | \alpha \neq 0 \wedge [\forall\beta\forall\gamma(\beta < \alpha \wedge \gamma < \alpha) \to \beta + \gamma < \alpha]\}$$

which is κ-club for all regular $\kappa > \omega$. Second, we need the concept of fix-points.

Definition 1.29. Assume $f : \mathbf{ON} \to \mathbf{ON}$ is order preserving. Then

$$Fix(f) := \{\alpha \in ON | f(\alpha) = \alpha\},$$

$$f' := en_{Fix(f)}.$$

Assume $\mathbf{A} \subset \mathbf{ON}$ is a class, then $\mathbf{A}' := Fix(en_{\mathbf{A}}) = \{\alpha \in \mathbf{A} | en_{\mathbf{A}}(\alpha) = \alpha\}$.

From that we can define the Veblen Hierarchy.

Definition 1.30. Veblen Hierarchy

- $Cr(0) := \mathbb{H}$
- $Cr(\alpha + 1) := Cr(\alpha)'$
- If β is a limit, then $Cr(\beta) := \bigcap\{Cr(\alpha) | \alpha < \beta\}$.

The function $\chi_\alpha := en_{Cr(\alpha)}$ is called the *Veblen function of* α.

Ordinals from $Cr(\alpha)$ are called α-critical ordinals. A very important class of ordinals are the epsilon numbers

$$\mathbb{E} := \{\alpha \in \mathbf{ON} | \omega^\alpha = \alpha\}$$

where the smallest of its elements is denoted by

$$\varepsilon_0 := \bigcap \mathbb{E}$$

which can be imagined by an infinite ω exponentiation. The next lemma shows that $\mathbb{E} = Cr(1)$.

Lemma 1.31. *Assume α and γ are ordinals. Then χ_α have the properties:*

1. $\chi_0(\alpha) = \omega^\alpha$
2. $\chi_1(0) = \varepsilon_0$
3. *If* $\beta < \alpha$*, then* $\chi_\gamma(\beta) < \chi_\gamma(\alpha)$*.*
4. $\beta \leq \chi_\alpha(\beta)$
5. *If* $\alpha < \beta$*, then* $Cr(\beta) \subsetneq Cr(\alpha)$ *and* $\chi_\alpha(\gamma) \leq \chi_\beta(\gamma)$ *and* $\chi_\alpha(\chi_\beta(\gamma)) = \chi_\beta(\gamma)$*.*

Proof. See [8, Ch.3, p.37]. $\square$

Theorem 1.32. *For all regular* $\kappa \in \boldsymbol{ON}$ *with* $\kappa > \alpha, \omega$ *holds that* $Cr(\alpha)$ *is* κ*-club.*

Proof. See [8, Ch.3, p.37] $\square$

The following theorem gives the first hint how a primitive recursive notation system of some ordinals can be defined.

Theorem 1.33. ***Basic Conditions of Veblen Functions***

- $\chi_{\alpha_1}(\beta_1) = \chi_{\alpha_2}(\beta_2)$ *iff one of the following conditions holds:*
  1. $\alpha_1 < \alpha_2$ *and* $\beta_1 = \chi_{\alpha_2}(\beta_2)$
  2. $\alpha_1 = \alpha_2$ *and* $\beta_1 = \beta_2$
  3. $\alpha_2 < \alpha_1$ *and* $\beta_2 = \chi_{\alpha_1}(\beta_1)$
- $\chi_{\alpha_1}(\beta_1) < \chi_{\alpha_2}(\beta_2)$ *iff one of the following conditions holds:*
  1. $\alpha_1 < \alpha_2$ *and* $\beta_1 < \chi_{\alpha_2}(\beta_2)$
  2. $\alpha_1 = \alpha_2$ *and* $\beta_1 < \beta_2$
  3. $\alpha_2 < \alpha_1$ *and* $\beta_2 > \chi_{\alpha_1}(\beta_1)$

Proof. See [8, Ch.3, p.38]. $\square$

The lambda notation will be used to denote functions.

Corollary 1.34. *The function* $\lambda\alpha.\chi_\alpha(0)$ *is order preserving, so for all* β *and* α *it holds that* $\alpha \leq \chi_\alpha(0) \leq \chi_\alpha(\beta)$*.*

Some of the critical ordinals are some kind of fix-points in this hierarchy and give important upper bounds for the use of ordinal notation systems.

Definition 1.35. The class of *strongly critical* ordinals is defined as follows:

$$\mathbf{SC} := \{\alpha \in \mathbf{ON} | \alpha \in Cr(\alpha)\}.$$

And we denote them with $\Gamma_\alpha := en_{\mathbf{SC}}(\alpha)$.

I will denote the 2-array function

$$\chi := \lambda\alpha\beta.\chi_\alpha(\beta)$$

with the term χ-function and the the function

$$\chi_{\mathsf{T}}(\alpha,\beta) := \begin{cases} 2^\beta & : \quad \alpha = 0 \\ \chi_\alpha(\beta) & : \quad \alpha \neq 0 \end{cases}$$

with the term χ_{T}-function. T stands for Tait, because W. W. Tait used this function to get bounds for the increasing of ranks when eliminating cuts in deductions(see Theorem 3.32). Both versions of the χ-function are essential the same (see Proposition 1.17) but χ_{T} is more useful in the finite. The following lemma will show how the χ-function is related to the strongly critical ordinals.

Lemma 1.36. *1. $\alpha \in \boldsymbol{SC}$ iff for all $\beta, \gamma < \alpha$ holds $\chi_\beta(\gamma) < \alpha$.*

2. $\alpha \in \boldsymbol{SC}$ iff $\chi_\alpha(0) = \alpha$.

Proof. See [8, Ch.3, p.39]. □

From the above lemma it is clear that a strongly critical ordinal is closed under the χ-function. From this it is possible to use a Γ_α as a ordinal system for proof theory. Now a last theorem about **SC** is given.

Theorem 1.37. *For all regular $\alpha \in \boldsymbol{ON}$ with $\alpha > \omega$ the class $\boldsymbol{SC}$ is α-club.*

Proof. See [8, Ch.3, p.40] □

To get a notation system a normal forms are useful. The next proposition will help to find a normal form for any ordinal up to Γ_0.

Proposition 1.38. *For all $\alpha \in \mathbb{H}\backslash\boldsymbol{SC}$ there are unique determined ordinals $\beta, \gamma < \alpha$ such that $\alpha = \chi_\beta(\gamma)$.*

Proof. See [8, Ch.3, p.41]. □

These leads to a new normal form concept.

Corollary 1.39. *Every $\alpha < \Gamma_0$ can be uniquely represented in the form $\alpha = \chi_{\beta_1}(\gamma_1) + ... + \chi_{\beta_n}(\gamma_n)$ for $n < \omega$ and $\beta_i, \gamma_i < \alpha$ for $1 \leq i \leq n$.*

Proof. From Theorem 1.18 and Lemma 1.31 follows that every $\alpha < \Gamma_0$ have a unique representations of a sum of ordinals from $\mathbb{H}$. Together with Proposition 1.38 the proof finished. □

1.2 An Ordinal Notation System

In this section I present a way to deal in a primitive recursive way with ordinals up to Γ_0. By Corollary 1.39 the notation system can be given in terms of finite sums (coded by "1") and the χ-function (coded by "2"). Note that "$=$" in the following definition means that two finite strings are identical.

Definition 1.40. Ordinal Notation
Fix the alphabet $\{\langle, \rangle, 0, 1, 2\}$. We will define a set **OT** of finite strings and the string relation "$\prec$" ("$\preceq$" denotes as usual "$=$ or $\prec$") by induction on the length of the strings.

1. $0 \in \mathbf{OT}$
2. If $a_1, ..., a_n \in \mathbf{OT}$ and $a_1 \succeq ... \succeq a_n$ then $\langle 1, a_1, ..., a_n \rangle \in \mathbf{OT}$.
3. If $a_1, a_2 \in \mathbf{OT}$ then $\langle 2, a_1, a_2 \rangle \in \mathbf{OT}$.
4. $0 \prec a$ for all $a \in \mathbf{OT}$ not identical to 0.
5. If $a, b \in \mathbf{OT}$ such that $a = \langle 2, a_1, a_2 \rangle$ and $b = \langle 2, b_1, b_2 \rangle$, then $a \prec b$ iff one of the following cases holds:
   (a) $a_1 \prec b_1$ and $a_2 \prec b$
   (b) $a_1 = b_1$ and $a_2 \prec b_2$
   (c) $b_1 \prec a_1$ and $b_2 \succ a$,
6. If $a, b \in \mathbf{OT}$ such that $a = \langle 1, a_1, ..., a_n \rangle$ and $b = \langle 1, b_1, ..., b_m \rangle$, then $a \prec b$ iff one of the following cases holds:
   (a) $n < m$ and $a_i = b_i$ for all $1 \leq i \leq n$.
   (b) There exists $j \leq \min\{n, m\}$ such that $a_j \prec b_j$ and $a_i = b_i$ for all $1 \leq i \leq j-1$.

We call the elements of **OT** *ordinal terms*.

The next theorem gives us the crucial property of this notation system.

Theorem 1.41. *The set **OT** and the relations "$\prec$" are primitive recursive.*

Proof. The proof is an easy induction on the length of the strings and is done in [12, p.87]. □

As promised at the beginning of this chapter we now show how the ordinal notation system is related to the ordinals given in **ZFC**. For this we define the canonical embedding ι on **OT** into **ON**. This embedding will prove that "$\prec$" is linear-ordered and well-founded. Note that one can prove most of the properties of the relation "$\prec$", except that it is well-founded, without referring to ordinals (see [12, Ch.V, §14]).

Remark 1.42. Note that this notation system satisfies the conditions of an elementary recursive ordinal notation system (see [4]).

Definition 1.43. The function $\iota : \mathbf{OT} \to \mathbf{ON}$ is defined as follows:

1. $\iota(0) := 0$
2. If $a = \langle 2, a_1, a_2 \rangle$ then $\iota(a) := \chi_{\iota(a_1)}(\iota(a_2))$.
3. If $a = \langle 1, b_1, ..., b_m \rangle$ then $\iota(a) := \iota(a_1) + ... + \iota(a_n)$.

From this definition and the previous section we get the next theorem.

Theorem 1.44. *For every $a \in \boldsymbol{OT}$ it holds that $\iota(a) < \Gamma_0$ and for every ordinal $\alpha < \Gamma_0$ there is a unique ordinal term $a \in \boldsymbol{OT}$ such that $\iota(a) = \alpha$. Moreover: $\iota(a) < \iota(b)$ iff $a \prec b$.*

Proof. This follows from Proposition 1.39 together with Definition 1.43 by induction on the length of strings and transfinite induction. $\square$

Remark 1.45. From the Definitions 1.40 and 1.43 together with Theorem 1.44 it should be clear what element of **OT** is denoted by a term of ordinal arithmetic. On the other hand many functions of ordinal arithmetic such as $+, \cdot$, exponentiation and the natural sum can be introduced as primitive recursive functions on **OT** (see [12, §14] and [12, §16]).

Definition 1.46. Assume $a \in \mathbf{OT}$, then $(\mathbf{OT}_a, \prec_a)$ denotes the ordinal notation system where

- $\mathbf{OT}_a := \{b \in \mathbf{OT} | b \prec a\}$
- "$\prec_a$" denotes the restriction of "$\prec$" to $\mathbf{OT}_a$

We will usually identify $a \in \mathbf{OT}$ and $\alpha \in \mathbf{ON}$ in this notation if $\iota(a) = \alpha$.

Remark 1.47. As in Remark 1.45 the functions of ordinal arithmetic can also introduced as primitive recursive functions on all the $(\mathbf{OT}_a, \prec_a)$.

Using the notation system we can talk about ordinal terms, the elements of **OT** or $\mathbf{OT}_\alpha$, instead of ordinals. Strings can be coded into the language of arithmetic as usual, e.g. via Gödel numbers. For example we can define a coding $\ulcorner . \urcorner : \mathbf{OT} \to \mathbb{N}$ as follows:

$$\ulcorner \langle \urcorner := 0$$

$$\ulcorner \rangle \urcorner := 1$$

$$\ulcorner 0 \urcorner := 2$$

$$\ulcorner 1 \urcorner := 3$$

$$\ulcorner 2 \urcorner := 4$$

$$\ulcorner a \urcorner := \prod_{i=1}^{n} p_i^{\ulcorner a_i \urcorner + 1}$$

where a is the string $a_1 ... a_n$ and p_i denotes the i-th prime number.

1.3 The Proof Theoretical Ordinal

The most used method in proof theory is induction over the complexity of formulas or deductions. As mentioned at the beginning of this chapter the usual way of expressing the rank of formulas and deductions via natural number is not suitable for many purposes. The notation system which is introduced in Definition 1.40 uses a wider system, which can also be coded into the language of arithmetic. We can formulate for an arbitrary formula φ the transfinite induction over $\alpha < \Gamma_0$ in the language of arithmetic, by coding the ordinal notation system $(\mathbf{OT}_\alpha, \prec_\alpha)$ into arithmetic, as follows: $\mathbf{TI}_\alpha(\varphi)$ is the following formula:

$$\forall x[\mathbf{OT}_\alpha(x) \to [\forall y \prec_\alpha x \to \varphi(y)] \to \varphi(x)] \to \forall x[\mathbf{OT}_\alpha(x) \to \varphi(x)].$$

The schema of α-induction is denoted by $\mathrm{TI}_\alpha := \{\mathbf{TI}_\alpha(\varphi) | \varphi \text{ is a formula}\}$. For a set of formulas Λ which appears in the arithmetical hierarchy (see [8]) we will denote the schema of Λ-α-induction by $\mathrm{TI}_{\alpha,\Lambda} := \{\mathbf{TI}_\alpha(\varphi) | \varphi \in \Lambda\}$.
By this observation it is enough to use an arithmetical system as framework for the following observations. A rather weak such framework is the theory of *primitive recursive arithmetic*, denoted by **PRA**, which includes the defining axioms for all primitive recursive functions (as universal sentences) and the usual induction schema restricted to quantifier-free formulas φ:

$$\varphi(0) \wedge \forall x[\varphi(x) \to \varphi(x+1)] \to \forall x \varphi(x).$$

From this we can define what we mean by the proof theoretic ordinal (as Michael Rathjen does in [9]) of a calculus $\mathbf{C}$. From now on we denote an ordinal and the corresponding ordinal term by the same small Greek latter.

Definition 1.48. Let $\mathbf{C}$ be a calculus defined by a finite set of deduction rules and a primitive recursive set of axioms in a primitive recursive language. Assume there is an $\alpha \in \mathbf{OT}$ such that

$$\mathbf{PRA} + \mathbf{TI}_{\alpha,\Delta_0} \vdash \mathrm{Con}(\mathbf{C}),$$

then we call the $\prec$-least of this α the *proof theoretical ordinal of* $\boldsymbol{C}$ and denote it by $\|\mathbf{C}\|_{\mathrm{Con}}$.

Remark 1.49. Such an element of $\mathbf{OT}$ does not have to exist. For example Kripke-Platek set theory (considered as a Tait-calculus) need an ordinal notation system which goes far beyond Γ_0.
As usual we code formulas and deductions into arithmetic to define:

$$\mathrm{Prf}_{\mathbf{C}}\, x :\Leftrightarrow \exists y\, \mathrm{Prf}_{\mathbf{C}}(y, x)$$

$$\mathrm{Con}(\mathbf{C}) :\Leftrightarrow \neg\, \mathrm{Prf}_{\mathbf{C}} \ulcorner \bot \urcorner,$$

where $\ulcorner \varphi \urcorner$ means the Gödel number of a formula φ and $\mathrm{Prf}_{\mathbf{C}}(y, x)$ a primitive recursive predicate means "y is the Gödel number of a deduction for a formula with Gödel number x".

As one can define $\|\mathbf{C}\|_{\text{Con}}$ with the help of the schema of transfinite induction over an ordering there is also a possible definition which use the related concept that an ordering is well-founded. Let R be a two place primitive recursive total order relation on the primitive recursive set A, then the schema $\mathbf{PRWO}(R)$ says that every primitive recursive definable R-descending sequence is finite. $\mathbf{PRWO}(R)$ is weaker than "R is a well-order" and can be formulated in arithmetic as

$$\forall\vec{x}\exists y[\neg[f(\vec{x}, y+1)Rf(\vec{x}, y)] \vee \neg A[f(\vec{x}, y)]] \wedge \mathbf{LO}(R)$$

where $\mathbf{LO}(R)$ means that (A, R) is a linear order, $f \in \mathcal{L}(\mathbf{PRA})$ and R, A abbreviations for the defining formulas of the relation and the set respectively.

Notation 1.50. In the rest of this diploma thesis (A, R) will always be a notation system $(\mathbf{OT}_\alpha, \prec_\alpha)$ for a $\alpha < \Gamma_0$. In this case we denote the schema defined above by $\mathbf{PRWO}(\alpha)$.

Definition 1.51. Let $\mathbf{C}$ be a calculus defined by a finite set of deduction rules and a primitive recursive set of axioms in a primitive recursive language. Assume there is an $\alpha \in \mathbf{OT}$ such that

$$\mathbf{PRA} + \mathbf{PRWO}(\alpha) \vdash \text{Con}(\mathbf{C}),$$

then we denote the $\prec$-least of this α by $\|\mathbf{C}\|_{\text{Con}}^{\text{PRWO}}$.

Michael Rathjen[1] points out that if $\alpha \in Cr(1)$ then $\mathbf{PRA} + \mathbf{TI}_{\alpha,\Delta_0}$ is contained in $\mathbf{PRA} + \mathbf{PRWO}(\alpha)$ (in the sense that every sentence which is provable in $\mathbf{PRA} + \mathbf{PRWO}(\alpha)$ is also provable in $\mathbf{PRA} + \mathbf{TI}_{\alpha,\Delta_0}$) and both theories prove the same Π^0_2-sentences.

As was pointed out by Kreisel, the restriction to a fixed notation system in both previous definitions is crucial. Defining the proof theoretical ordinal independently of a fixed notation system is not possible, i.e. we can not define $\|\mathbf{C}\|_{\text{Con}}$ or $\|\mathbf{C}\|_{\text{Con}}^{\text{PRWO}}$ as the order-type of the shortest primitive recursive ordering which is needed to prove the consistency. We will demonstrate this in the following:
Recall that $\|\mathbf{PA}\|_{\text{Con}} = \varepsilon_0$ (see [8, p.126]) and $\|\mathbf{PA}\|_{\text{Con}}^{\text{PRWO}} = \varepsilon_0$ (see [16]) and note that we can apply the following theorem to $\mathbf{PA}$.

Theorem 1.52. ***Kreisel [8]***
Let $\boldsymbol{C}$ be a calculus defined by a finite set of deduction rules and a primitive recursive set of axioms in a primitive recursive language. Then there is a primitive recursive well order $\prec_{\boldsymbol{C}}$ (on $\mathbb{N} \times \mathbb{N}$) with $\text{otyp}(\prec_{\boldsymbol{C}}) = \omega$ such that

$$\boldsymbol{PRA} + \boldsymbol{PRWO}(\prec_{\boldsymbol{C}}) \vdash \text{Con}(\boldsymbol{C}).$$

[1] personal communication, 5th June 2011

Proof. At first we define the relation:

$$x \prec_{\mathbf{C}} y :\Leftrightarrow \begin{cases} x < y & \text{if} \quad (\forall i < x)[\neg \operatorname{Prf}_{\mathbf{C}}(i, \ulcorner \bot \urcorner)] \\ y < x & \text{if} \quad \text{otherwise} \end{cases}$$

which is primitive recursive and a total order.
We define $\varphi(x) :\equiv (\forall i \leq x)[\neg \operatorname{Prf}_{\mathbf{C}}(i, \ulcorner \bot \urcorner)]$.
So we get

$$(*) : \mathbf{PRA} \vdash (\forall x \prec_{\mathbf{C}} y)\varphi(x) \to \varphi(y),$$

because if we have $\neg\varphi(y)$ then by definition of the formula we obtain

$$(\exists i \leq y)[\operatorname{Prf}_{\mathbf{C}}(i, \ulcorner \bot \urcorner)]$$

and because $y < y + 1$ we get

$$(\exists i \leq y + 1)[\operatorname{Prf}_{\mathbf{C}}(i, \ulcorner \bot \urcorner)].$$

This means by definition of $\prec_{\mathbf{C}}$ that $y < y+1$, which is in this case equivalent to $y + 1 \prec_{\mathbf{C}} y$. But with the assumption of the statement $(*)$, that for all $x \prec_{\mathbf{C}} y$ we have $\varphi(x)$, we also get $\varphi(y + 1)$, which is by definition

$$(\forall i \leq y + 1)[\neg \operatorname{Prf}_{\mathbf{C}}(i, \ulcorner \bot \urcorner)].$$

This leads to the weaker statement

$$(\forall i \leq y)[\neg \operatorname{Prf}_{\mathbf{C}}(i, \ulcorner \bot \urcorner)],$$

which is again by definition just $\varphi(y)$.
Altogether, we get $\neg\varphi(y) \Rightarrow \varphi(y)$, with is a contradiction. That leads to $\varphi(y)$, which proves $(*)$.
Now $(*)$ implies

$$\mathbf{PRA} + \mathbf{PRWO}(\prec_{\mathbf{C}}) \vdash \forall x \varphi(x),$$

which leads with the definition of $\operatorname{Con}(\mathbf{C})$ to the statement

$$\mathbf{PRA} + \mathbf{PRWO}(\prec_{\mathbf{C}}) \vdash \operatorname{Con}(\mathbf{C}).$$

It remains to show that $\operatorname{otyp}(\prec_{\mathbf{C}}) = \omega$. Because of the assumption that $\mathbf{C}$ is consistent the sentence $\operatorname{Con}(\mathbf{C})$ is true and so the statement $(\forall i < x)[\neg \operatorname{Prf}_{\mathbf{C}}(i, \ulcorner \bot \urcorner)]$ holds for every x. This means by definition of $\prec_{\mathbf{C}}$ the relation is just the standard ordering of natural numbers $<$, which is a well ordering of order type ω. □

This shows that the proof theoretical ordinals only make sense with respect to a specified notation system, and not as ordinals in the set theoretical sense. However by leaving the realms of consistency there are two notations of proof theoretical ordinals which are more robust then the two previously mentioned: $\|T\|_{\text{otyp}}$ and $\|T\|_{\Pi^1_1}$ for suitable theories T. We give a definition of $\|T\|_{\text{otyp}}$ here (for another one, see Definition 3.51), the definition of $\|T\|_{\Pi^1_1}$ will be given in Definition 3.51.

Definition 1.53. Assume T is a first order theory which is strong enough to code primitive recursive sets. Then set $\|T\|_{\mathrm{otyp}}$ to be the supremum of all $\mathrm{otyp}(\prec)$ where $(A, \prec)$ is primitive recursive and T proves all instances of $\mathbf{PRWO}(\prec)$.

Remark 1.54. Note $\|T\|_{\mathrm{otyp}}$ is really an ordinal while $\|\mathbf{C}\|_{\mathrm{Con}}$ and $\|\mathbf{C}\|_{\mathrm{Con}}^{\mathrm{PRWO}}$ are elements of the used notation system.

2 The Consistency Proof by Gerhard Gentzen

In this chapter I will present Gentzens consistency proof [16] for the following, slightly generalized, situation: The considered theories are axiomaticed by universal sentences plus induction axioms of the form:

$$\varphi(c_1) \wedge ... \wedge \varphi(c_m) \wedge \forall \vec{x}[\varphi(x_1) \wedge ... \wedge \varphi(x_n) \to \varphi(f(x_1, ..., x_n))] \to \forall x \varphi(x)$$

where f is a function such that all closed terms of the language are provable equal to a term build up from $c_1, ..., c_m$ and f.

Remark 2.1. W.l.o.g. we use only one function. Note that one can build from a finite number of functions a single function which can be split into its parts.

The proof proceeds as follows:

1. The considered theory T_f is translated into a sequent calculus $\mathbf{CT}_f$. In this translation, the induction axioms will be replaced by a deduction rule.

2. A weak subcalculus $\mathbf{SCT}_f$, called the simple part of $\mathbf{CT}_f$, will be defined. We will have to assume the consistency of $\mathbf{SCT}_f$ and that $\mathbf{SCT}_f$ proves for every closed term t the equality of t to a term $\bar{t}$ build up just from $c_1, ..., c_m$ and f (we call this $\bar{t}$ an f-term).

3. We assign to each deduction D of $\mathbf{CT}_f$ an ordinal term (the rank of D), denoted by $o(D)$.

4. We define the end-piece of a deduction D as the part in which no conclusion of a logical inference (which is also in this part) vanishes by an application of the cut rule.

5. Now assume towards a contradiction that $\mathbf{CT}_f$ is inconsistent. Show that for every deduction D, which is not in the simple part, if it is a deduction of an inconsistency then, there is another deduction D' of an inconsistency such that $o(D') \prec o(D)$. Moreover we can construct D by the following primitive recursive operations on the end-piece of D:

   (a) Eliminate all disruptive factors from the end-piece: free variables, application of induction, logical axioms ore weakening rules. (Here we use properties of $\mathbf{SCT}_f$ which are guaranteed by the fact that all axioms of T_f are universal sentences and that $\mathbf{SCT}_f$ satisfies the condition assumed in 2.)

   (b) Make a partial cut-elimination to get a deduction of an inconsistency where the cut formulas have smaller ranks then the cut formulas of the eliminated cuts and therefore are "nearer" to the simple part of the calculus.

6. Using the fact that the ordinal notation system $(\mathbf{OT}_{\varepsilon_0}, \prec_{\varepsilon_0})$ is well-founded, we show that the iteration of the method described in 5 terminates and leads to a deduction of a contradiction in the simple part. This contradicts the assumption in 2.

7. As an additional step one can check that every step in the proof, except the termination, can be done in **PRA** providing that **PRA** already proves that the assumptions in 2 holds. So in this case we can prove that the consistency of $\mathbf{SCT}_f$ imply the consistency of $\mathbf{CT}_f$ in $\mathbf{PRA}$+$\mathbf{PRWO}(\varepsilon_0)$.

2.1 Language and Theory

For a set τ of constants, we define the first order languages $\mathcal{L}(\tau)$.

Definition 2.2. The *primitive symbols* are:

1. Logical Symbols
   (a) Logical connectives: $\wedge, \vee, \rightarrow, \neg$
   (b) Quantifiers: $\exists, \forall$
   (c) Free variables: $a_0, a_1, a_2, ...$
   (d) Bound variables: $x_0, x_1, x_2, ...$
   (e) Brackets: (,)

2. Constants: The set τ consists finitely or countable many of the following constants:
   (a) Individual constants: $c_1, c_2, c_3, ...$
   (b) Function constants for n-array function with $n \in \omega$:
   $f_1^n, f_2^n, f_3^n, ...$
   (c) One 2-array Predicate constant: $=$
   (d) Predicate constants for n-array Predicates with $n \in \omega$:
   $P_1^n, P_2^n, P_3^n, ...$

I will use latin letters like a, b, c as metavariables for free, x, y, z as metavariables for bound variables, f, h, g as metavariables for function constants and P, R as metavariables for predicate constants. Finite sequences of primitive symbols will be called expressions.

Definition 2.3. The set of τ-*terms* (or just terms) is defined recursively:

1. Each free variable and each individual constant is a term.

2. If f is a n-array function constant and $t_1, ..., t_n$ are terms then $f(t_1, ...t_n)$ is a term.

Terms without free variables will be called *closed terms.*

We will use the symbol "$\equiv$" as a meta symbol for identity over expressions.

Definition 2.4. The expression φ is an *atomic formula* iff $\varphi \equiv = (t_1, t_2)$ ore $\varphi \equiv P(t_1, ..., t_n)$ where $t_1, ..., t_n$ are terms and P is an n-array predicate constant.

We will denote $= (t_1, t_2)$ by $t_1 = t_2$ from now on. Before we can define what formula means we have to introduce the notation of substitution.

Definition 2.5. Assume $\varphi, \sigma_1, ...\sigma_n$ are expressions and $\rho_1, ..., \rho_n$ distinct primitive symbols, then

$$\varphi[\rho_1, ..., \rho_n/\sigma_1, ..., \sigma_n]$$

denotes the expression where in every occurrence of ρ_i, for $1 \leq i \leq n$, the expression σ_i is written instead.

It should be clear that if $\rho_1, ..., \rho_n$ are distinct primitive symbols not occurring in φ then,

$$(\varphi[\rho_1, ..., \rho_n/\sigma_1, ...\sigma_n])[\sigma_1, ...\sigma_n/\theta_1, ..., \theta_n]$$

is the same as

$$\varphi[\rho_1, ..., \rho_n/\theta_1, ..., \theta_n].$$

Definition 2.6. The set $\mathcal{L}(\tau)$ is defined inductively:

1. If φ is an atomic formula, then $\varphi \in \mathcal{L}(\tau)$.
2. If $\varphi, \psi \in \mathcal{L}(\tau)$ then $(\neg\varphi), (\varphi \wedge \psi), (\varphi \vee \psi), (\varphi \to \psi) \in \mathcal{L}(\tau)$.
3. If $\varphi(a) \in \mathcal{L}(\tau)$ and the bound variable x does not occur in $\varphi(a)$, then $\exists x(\varphi(x)), \forall x(\varphi(x)) \in \mathcal{L}(\tau)$ where $\varphi(x)$ denotes $\varphi(a)[a/x]$.

Now some standard definitions for languages are given.

Definition 2.7. 1. Elements of $\mathcal{L}(\tau)$ will be called *formulas.*

2. Elements of $\mathcal{L}(\tau)$ without quantifiers are called *quantifier free formulas.*
3. Elements of $\mathcal{L}(\tau)$ without free variables will be called *sentences.*
4. A set of formulas will be called a τ*-Theory.*

The next definition gives a useful subset of $\mathcal{L}(\tau)$.

Definition 2.8. A formula φ is in prenex normalform iff $\varphi \equiv Q_1 x_1, ..., Q_n x_n \psi(x_1, ..., x_n)$ where $\psi(a_1, ..., a_n)$ is quantifier free and $Q_i \in \{\forall, \exists\}$ for all $1 \leq i \leq n$. The formula $\psi(a_1, ..., a_n)$ is called the *matrix* of φ.

This definition leads to a well known fact.

Proposition 2.9. *Every formula is logical equivalent to a formula in prenex normal form.*

Proof. See [2, p.160]. □

Remark 2.10. By logical equivalence of two formulas, φ and ψ, one means that a complete and sound first order calculus deducts the formula $\varphi \leftrightarrow \psi$, or equivalently, this formula is valid in the semantical sense [2].

The next two definitions give also important subsets of $\mathcal{L}(\tau)$.

Definition 2.11. A sentence φ is called a *universal sentence* iff φ is in prenex normal form and $\exists$ does not occur in φ.

Definition 2.12. Assume φ is quantifier free.
Say that φ is in *conjunctive normal form* iff

$$\varphi \equiv \bigwedge_{i=1}^{n} \bigvee_{j=1}^{k_i} \psi_{ij}$$

and the ψ_{ij} are atomic or negations of atomic formulas. An universal sentence is in constructive normal form iff its matrix is in constructive normal form.

These definition leads to the next well known fact.

Proposition 2.13. *Every quantifier free formula is equivalent to a formula in conjunctive normal form.*

Proof. See [2, p.53]. □

Definition 2.14. Two theories T and T' are *logical equivalent* iff T and T' prove exactly the same sentences.

From this results we can prove the next lemma.

Lemma 2.15. *Assume T is a theory containing only universal sentences $\varphi \equiv \forall\vec{x}\psi(\vec{x})$ where the conjunctive normal form of $\psi(\vec{a})$ is*

$$\bigwedge_{i=1}^{n} \bigvee_{j=1}^{k_i} \psi_{ij}$$

. Let T' is the theory containing for all $\varphi \in T$ and all $1 \leq i \leq n$ the sentence

$$\forall\vec{x}(\bigvee_{j=1}^{k_i} \psi_{ij}),$$

then is T' logical equivalent to T.

Proof. From Proposition 2.13 follows that $\forall\vec{x}\psi(\vec{x})$ is logical equivalent to $\forall\vec{x}(\bigwedge_{i=1}^{n}\bigvee_{j=1}^{k_i}\psi_{ij})$ which is logical equivalent to $\bigwedge_{i=1}^{n}\forall\vec{x}(\bigvee_{j=1}^{k_i}\psi_{ij})$, because there are only universal quantifiers. It is easy to see that $\{\bigwedge_{i=1}^{n}\forall\vec{x}(\bigvee_{j=1}^{k_i}\psi_{ij})\}$ is logical equivalent to $\{\forall\vec{x}(\bigvee_{j=1}^{k_1}\psi_{1j}),...,\forall\vec{x}(\bigvee_{j=1}^{k_n}\psi_{nj})\}$. □

Definition 2.16. Let $f, c_1, ..., c_m \in \tau$. A formula which is an instance of the schema

$$\varphi(c_1) \land ... \land \varphi(c_m) \land \forall\vec{x}[\varphi(x_1) \land ... \land \varphi(x_n) \to \varphi(f(x_1, ..., x_n))] \to \forall x\varphi(x)$$

where $\varphi(a) \in \mathcal{L}(\tau)$ will be called a *induction axiom for f and $c_1, ..., c_m$*.

Notation 2.17. Assume f is a function constant and $c_1, ..., c_m$ are individual constants occurring in the induction axioms of a Theory T. Terms which are build up from $c_1, ..., c_m$ and f only are called *f-terms of T*.

Now we make the assumptions on the considered theories T_f explicit.

Assumtion 2.18. Assume τ is a primitive recursive set of constants. Let f be a function constant and $c_1, ..., c_m$ individual constants of τ. We assume $T_f = (T_f)_0 \cup (T_f)_{\text{Ind}}$ is a primitive recursive τ-theory which satisfies the following conditions:

1. $(T_f)_{\text{Ind}}$ consists of all induction axioms for f and $c_1, ..., c_m$.

2. Every element of $(T_f)_0$ is a universal sentence. According to Lemma 2.15 we can assume with out lose of generality that the elements of $(T_f)_0$ are of the form
$$\forall\vec{x}[\varphi_1(\vec{x}) \lor ... \lor \varphi_n(\vec{x})]$$
where the $\varphi_i(\vec{a})$ for $1 \le i \le n$ are atomic or the negotiations of atomic formulas.

3. For all closed τ-terms t there is an f-term $\bar{t}$ such that $T_f \vdash t = \bar{t}$. Moreover: all individual and function constance of τ occur in a sentence of T_f.

Remark 2.19. Condition 3 is actually not sufficient for the proof. It will be strengthened later such that already the simple part proves the identity.

Notation 2.20. In general we denote the induction free part of an theory T by T_0.

2.2 Some Sequent Calculi

After we know what we mean by a formula we can define what we mean by a sequent calculus and from that we will define the sequent calculus of a given theory T_f. For this I will introduce the central syntactical object of a sequent calculus, the sequent.

Definition 2.21. A *sequent* $\mathcal{S}$ is a expression of the form $\Gamma \Rightarrow \Delta$ where Γ and Δ are finite sequences of formulas.

For an intuitive understanding of what a sequent means, one can interpret it as the formula

$$\bigwedge_{\varphi\in\Gamma} \varphi \to \bigvee_{\psi\in\Delta} \psi$$

where $\bigwedge\{\phi_1, ...\phi_n\}$ means $\phi_1 \wedge ... \wedge \phi_n$ and vice versa for "$\bigvee$". Equivalent, Γ semantically implies the disjunction of Δ, in symbols

$$\Gamma \models \bigvee \Delta.$$

If one of the sequences is empty than it is omitted. Semantically $\bigwedge\emptyset$ is interpreted as $\top$ and $\bigvee\emptyset$ as $\bot$. The sequent where both sequences are empty is denoted by $\Rightarrow$ and is called the *empty-sequent* i.e., $\Rightarrow$ is a contradiction. The notation Γ, φ denotes the sequence starting with Γ as subsequence and ends with φ.

Definition 2.22. An *inference* is a tuple of the form

$$\langle \mathcal{S}, \langle \mathcal{S}_i | i \in I\rangle\rangle$$

where $\mathcal{S}$ and $\mathcal{S}_i$ for $i \in I$ are sequents and I is a finite set. Fore easier reading we will denote them by

$$\mathcal{S} \text{ or } \frac{\mathcal{S}_0}{\mathcal{S}} \text{ or } \frac{\mathcal{S}_0 \quad \mathcal{S}_1}{\mathcal{S}}.$$

The $\mathcal{S}_i$ for $i \in I$ called *upper sequents* and $\mathcal{S}$ *lower sequent*. An inference without an upper sequent is called an *Axiom*. A *sequent calculus* (or just calculus) is a set of such inferences.

A sequent calculus will be allays defined by a finite set of *deduction rules* which are schemas for inferences. And we will always talk about the deduction rule when we talk about an inference which is an instance of this deduction rule.

From this definition we can introduce the concept of deduction in a calculus.

Definition 2.23. Assume $\mathbf{C}$ is a sequent calculus defined by a finite set of deduction rules. A *deduction* D of $\mathcal{S}$ in $\mathbf{C}$, $D \vdash \mathcal{S}$, is a finite, rooted, ordered tree[2], whose nodes are labelled by a sequent and (a tag of) a deduction rule, satisfying the following conditions:

1. All leaves are labelled by axioms and the corresponding axiom deduction rule.

[2] If two notes are comparable, then we call the note closer to the root "below" or "successor" of the other.

2. If a node, which is not a leaf, is labelled by a sequent $\mathcal{S}$ and a deduction rule R and if its immediate predecessors are labelled by the sequents $\mathcal{S}_1, ..., \mathcal{S}_n$, then the inference

$$\frac{\mathcal{S}_1 \quad ... \quad \mathcal{S}_n}{\mathcal{S}}$$

is an instance of the deduction rule R.

3. The root of D is labelled by $\mathcal{S}$ ($\mathcal{S}$ is called the *end-sequent*) and a deduction rule R (R is called the *last deduction rule*).

We say $\mathbf{C}$ derives a Sequent $\mathcal{S}$ (or a formula φ), in symbols $\mathbf{C} \vdash \mathcal{S}$ (or $\mathbf{C} \vdash \varphi$), if there is a deduction D in $\mathbf{C}$ such that $D \vdash \mathcal{S}$ (or $D \vdash \Rightarrow \varphi$).

We identify a note of the deductions with its label. With the previous work we can define the well known Gentzen calculus of classical logic **LK**.

Definition 2.24. *The Calculus **LK***[16]

1. Logical Axioms:

$$\varphi \Rightarrow \varphi$$

for every atomic $\varphi \in \mathcal{L}(\tau)$.

2. Structural rules:

(a) Weakening:

$$\frac{\Gamma \Rightarrow \Delta}{\varphi, \Gamma \Rightarrow \Delta} \text{(left)} \quad \frac{\Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta, \varphi} \text{(right)}$$

The formula φ is called the weakening-formula.

(b) Contraction:

$$\frac{\varphi, \varphi, \Gamma \Rightarrow \Delta}{\varphi, \Gamma \Rightarrow \Delta} \text{(left)} \quad \frac{\Gamma \Rightarrow \Delta, \varphi, \varphi}{\Gamma \Rightarrow \Delta, \varphi} \text{(right)}$$

(c) Exchange:

$$\frac{\Gamma, \psi, \varphi, \Pi \Rightarrow \Delta}{\Gamma, \varphi, \psi, \Pi \Rightarrow \Delta} \text{(left)} \quad \frac{\Gamma \Rightarrow \Delta, \psi, \varphi, \Lambda}{\Gamma \Rightarrow \Delta, \varphi, \psi, \Lambda} \text{(right)}$$

(d) Cut:

$$\frac{\Gamma \Rightarrow \Delta, \varphi \quad \varphi, \Pi \Rightarrow \Lambda}{\Gamma, \Pi \Rightarrow \Delta, \Lambda}$$

The formula φ is called the *cut-formula*. The rules (a)-(c) are called *weak structural rules*.

3. Logical Rules:

   (a) Negation:

$$\frac{(\neg\varphi), \Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta, \varphi}\ (\neg\text{-left}) \quad \frac{\Gamma \Rightarrow \Delta, (\neg\varphi)}{\varphi, \Gamma \Rightarrow \Delta}\ (\neg\text{-right})$$

   (b) Conjunction:

$$\frac{\varphi, \Gamma \Rightarrow \Delta}{(\varphi \wedge \psi), \Gamma \Rightarrow \Delta}\ (\wedge\text{-left}_1) \quad \frac{\varphi, \Gamma \Rightarrow \Delta}{(\psi \wedge \varphi), \Gamma \Rightarrow \Delta}\ (\wedge\text{-left}_2)$$

   and

$$\frac{\Gamma \Rightarrow \Delta, \varphi \quad \Gamma \Rightarrow \Delta, \psi}{\Gamma \Rightarrow \Delta, (\varphi \wedge \psi)}\ (\wedge\text{-right})$$

   (c) Disjunction:

$$\frac{\varphi, \Gamma \Rightarrow \Delta \quad \psi, \Gamma \Rightarrow \Delta}{(\varphi \vee \psi), \Gamma \Rightarrow \Delta}\ (\vee\text{-left})$$

$$\frac{\Gamma \Rightarrow \Delta, \varphi}{\Gamma \Rightarrow \Delta, (\varphi \vee \psi)}\ (\vee\text{-right}_1) \quad \text{and} \quad \frac{\Gamma \Rightarrow \Delta, \psi}{\Gamma \Rightarrow \Delta, (\varphi \vee \psi)}\ (\vee\text{-right}_2)$$

   (d) Implication:

$$\frac{\Gamma \Rightarrow \Delta, \varphi \quad \psi, \Pi \Rightarrow \Lambda}{(\varphi \to \psi), \Gamma, \Pi \Rightarrow \Delta, \Lambda}\ (\to\text{-left}) \quad \frac{\varphi, \Gamma \Rightarrow \Delta, \psi}{\Gamma \Rightarrow \Delta, (\varphi \to \psi)}\ (\to\text{-right})$$

   (e) Generalisation:

$$\frac{\varphi(t), \Gamma \Rightarrow \Delta}{\forall x \varphi(x), \Gamma \Rightarrow \Delta}\ (\forall\text{-left}) \quad \frac{\Gamma \Rightarrow \Delta, \varphi(a)}{\Gamma \Rightarrow \Delta, \forall x \varphi(x)}\ (\forall\text{-right})$$

   Where t is a term and a is a free variable not occurring in the lower sequent which is called the *eigenvariable*.

   (f) Existence:

$$\frac{\varphi(a), \Gamma \Rightarrow \Delta}{\exists x \varphi(x), \Gamma \Rightarrow \Delta}\ (\exists\text{-left}) \quad \frac{\Gamma \Rightarrow \Delta, \varphi(t)}{\Gamma \Rightarrow \Delta, \exists x \varphi(x)}\ (\exists\text{-right})$$

   Where t is a term and a is a free variable not occurring in the lower sequent which is called the *eigenvariable*.

The Rules (a)-(d) called propositional rules and (d) and (f) quantifier rules. $\forall$-right and $\exists$-left called strong the other cases weak quantifier rules.

The formulas in the upper sequents which are used in the rule called *auxiliary formulas* (a.f.) the formulas in the lower sequent which are modified called *principal formulas* (p.f.) and formulas which are not used called *side formulas* (s.f.).

Remark 2.25. Adding a set of axioms A to **LK** means to enlarge the set of axiom inferences by the sequent $\Rightarrow \varphi$ for all $\varphi \in A$. Note that A have to be a set of sentences (see Theorem 2.29).

A useful definition by dealing with cuts is the following.

Definition 2.26. A cut is called *inessential* iff the cut formula is atomic. Otherwise it is called *essential*.

LK have a very important and well known fact about the application of the cut rule which will be not proved here but is given for completeness.

Theorem 2.27. *Cut Elimination*
*Every sequent $\mathcal{S}$ with **LK** $\vdash \mathcal{S}$ has a deduction in **LK** without the application of a cut.*

Proof. See [16, §5]. □

Definition 2.28. A formula φ is called *valid* iff φ is true in every Model.

To argue that the given sequent calculi are equivalent to the T_f with the Hilbert calculus we formulate the completeness theorem for **LK**.

Theorem 2.29. *Completeness Theorem*
LK$\vdash \varphi$ *iff* φ *is valid.*
*Moreover: If A is a set of sentences as in Remark 2.25, then **LK** $+ A \vdash \varphi$ iff $A \models \varphi$ (A semantically implies φ).*

Proof. See [16, §8]. □

Remark 2.30. Note that here we need that A does not contain free variables, otherwise e.g. ($\forall$-right) might lead to a contradiction.

Now we define the simple part of a theory T_f.

Definition 2.31. The *simple calculus* for T_f, called $\mathbf{SCT}_f$, is defined as follows:

1. Identity Axioms: Let $t_1, ..., t_n, t'_1, ..., t'_n$ be closed terms and $f, P \in \tau$ a n-array function and predicate constant, then

$$t_1 = t_2, t_2 = t_3 \Rightarrow t_1 = t_3$$
$$t_1 = t'_1, ..., t_n = t'_n \Rightarrow f(t_1, ..., t_n) = f(t'_1, ..., t'_n)$$
$$t_1 = t'_1, ..., t_n = t'_n, P(t_1, ..., t_n) \Rightarrow P(t'_1, ..., t'_n)$$

   are axioms of $\mathbf{SCT}_f$.

2. Closed Term-T_f-Axioms: Assume $\forall\vec{x}\varphi(\vec{x}) \in (T_f)_0$ where (according to Assumption 2.18)

$$\varphi(\vec{x}) \equiv \varphi_1(\vec{x}) \vee ... \vee \varphi_n(\vec{x})$$

such that the $\varphi_i(\vec{a})$ are atomic or the negation of atomic formulas. Assume

$$\neg\psi_1(\vec{a}), ..., \neg\psi_l(\vec{a}), \psi_{l+1}(\vec{a}), ..., \psi_n(\vec{a})$$

is a list of the $\varphi_i(\vec{a})$ where all ψ_j for $1 \leq j \leq n$ are atomic formulas. Then the following sequent is an axiom of $\mathbf{SCT}_f$:

$$\psi_1(\vec{t}), ..., \psi_l(\vec{t}) \Rightarrow \psi_{l+1}(\vec{t}), ..., \psi_n(\vec{t})$$

where $\psi_j(\vec{t}) \equiv \psi_j(\vec{a})[\vec{a}/\vec{t}]$ and $t_1, ..., t_n$ are arbitrary **closed** terms in $\mathcal{L}(\tau)$.

3. All weak structural rules as deduction rules.

4. Inessential cuts as deduction rule.

The next lemma is obvious from the definition of the simple part and Theorem 2.29.

Lemma 2.32. *If* $\boldsymbol{SCT_f} \vdash \Gamma \Rightarrow \Delta$ *then* $(T_f)_0 \models \bigwedge \Gamma \to \bigvee \Delta$.

Now we strengthen Assumption 2.18.

Assumtion 2.33. The τ-theory T_f is as supposed in Assumption 2.18 but replace case 3 of Assumption 2.18 by the stronger condition:

For all closed τ-terms t there is an f-term $\bar{t}$ such that $\mathbf{SCT}_f \vdash t = \bar{t}$.

Remark 2.34. Note that this assumption generally excludes the use of Skolem-function.

Examples of T_f are given in Section 2.5.

Now the definition of the sequent calculus related to a τ-theory T_f will be given. Note that the induction axioms of T_f vanish and are replaced by a deduction rule.

Definition 2.35. *The Calculus to the Theory T_f ($\boldsymbol{CT_f}$)*
$\mathbf{CT}_f$ is defined as **LK** together with the following deduction rules.

1. Identity Axioms: Let $t_1, ..., t_n, t'_1, ..., t'_n$ be terms and $f, P \in \tau$ a n-array function and predicate constant, then

$$t_1 = t_2, t_2 = t_3 \Rightarrow t_1 = t_3$$
$$t_1 = t'_1, ..., t_n = t'_n \Rightarrow f(t_1, ..., t_n) = f(t'_1, ..., t'_n)$$
$$t_1 = t'_1, ..., t_n = t'_n, P(t_1, ..., t_n) \Rightarrow P(t'_1, ..., t'_n)$$

are axioms of $\mathbf{CT}_f$.

2. T_f-Axioms:(Same as in $\mathbf{SCT}_f$ but without the restriction to closed terms.) Assume $\forall\vec{x}\varphi(\vec{x}) \in (T_f)_0$ where (according to Assumption 2.18)

$$\varphi(\vec{x}) \equiv \varphi_1(\vec{x}) \vee ... \vee \varphi_n(\vec{x})$$

such that the $\varphi_i(\vec{a})$ are atomic or the negation of atomic formulas. Assume

$$\neg\psi_1(\vec{a}), ..., \neg\psi_l(\vec{a}), \psi_{l+1}(\vec{a}), ..., \psi_n(\vec{a})$$

is a list of the $\varphi_i(\vec{a})$ where all ψ_j for $1 \leq j \leq n$ are atomic formulas. Then the following sequent is an axiom of $\mathbf{CT}_f$:

$$\psi_1(\vec{t}), ..., \psi_l(\vec{t}) \Rightarrow \psi_{l+1}(\vec{t}), ..., \psi_n(\vec{t})$$

where $\psi_j(\vec{t}) \equiv \psi_j(\vec{a})[\vec{a}/\vec{t}]$ and $t_1, ..., t_n$ are arbitrary terms in $\mathcal{L}(\tau)$.

3. Induction for f (IND_f):

$$\frac{\varphi(a_1), ..., \varphi(a_n), \Gamma \Rightarrow \Delta, \varphi(f(a_1, ..., a_n))}{\varphi(c_1), ..., \varphi(c_m), \Gamma \Rightarrow \Delta, \varphi(t)}$$

where t is a arbitrary term in $\mathcal{L}(\tau)$. The free variables $a_1, ..., a_n$ will be called the *eigenvariables* of IND_f, and $\varphi(t)$ the *induction formula.*

Remark 2.36. 1. With respect to Remark 2.25 it is not clear why $\mathbf{CT}_f$ still proves the same formulas as T_f. By ($\forall$-right) and Theorem 2.29 it is equivalent to add $\Rightarrow \forall\vec{x}\varphi(\vec{x})$ or $\Rightarrow \varphi(\vec{a})$ as a new axiom. We chose to use axioms with free variables instead of sentences for technical reasons. (We can also add rules for substitution which would make the proof more complicated.)
By ($\forall$-right) it is also easy to see why adding the rule IND_f is equivalent to adding the induction formulas as axioms. Since $(T_f)_0$ contains universal sentences only, T_f together with the Hilbert calculus is equivalent to $\mathbf{CT}_f$. So a consistency proof of $\mathbf{CT}_f$ is a consistency proof of T_f.

2. It should be clear that $\mathbf{SCT}_f$ is a proper subsystem of $\mathbf{CT}_f$ in the sense that every deduction of $\mathbf{SCT}_f$ is a deduction of $\mathbf{CT}_f$, but the converse is false, because in general even logical axioms have no deductions in $\mathbf{SCT}_f$.

For short I will called the identity axioms and the T_f-axioms of $\mathbf{CT}_f$ *mathematical axioms.*

2.3 Important definitions and facts about $\mathbf{CT}_f$

In this section I will give some well known and important facts about calculi like $\mathbf{CT}_f$ and useful definitions for the following sections.

Definition 2.37. 1. Fix a deduction D. The *successor* of a formula φ is defined as follows:

   (a) If φ is a cut formula, then φ has no successor.

   (b) If φ is an auxiliary formula of an deduction rule other than a cut or exchange, then the principal formula is the successor of φ.

   (c) If φ is a auxiliary formula of exchange then φ in the lower sequent of exchange is the successor of φ.

   (d) If φ is the k-th formula of Γ, Π, Δ or Λ in the upper sequence, then the k-th formula of Γ, Π, Δ or Λ in the lower sequent is the successor of φ.

2. A sequence of sequents will be called a *thread* in a deduction D if the following properties are satisfied:

   (a) The sequence begin with an (logical or mathematical) axiom and ends with the end-sequent of D.

   (b) Every sequent in the sequence expect the last is an upper sequent of an inference, and is immediately followed by the lower sequent of this inference.

3. Assume $\mathcal{S}_1, \mathcal{S}_2$ and $\mathcal{S}_3$ are sequents in a deduction D. The sequent $\mathcal{S}_1$ is *above* $\mathcal{S}_2$ (or $\mathcal{S}_2$ is *below* $\mathcal{S}_1$) iff there is a thread containing $\mathcal{S}_1$ and $\mathcal{S}_2$ and $\mathcal{S}_1$ appears before $\mathcal{S}_2$. The sequent $\mathcal{S}_3$ is *between* $\mathcal{S}_1$ and $\mathcal{S}_2$ iff $\mathcal{S}_1$ is above $\mathcal{S}_3$ and $\mathcal{S}_2$ is below $\mathcal{S}_3$.

4. An inference is *below a sequent* $\mathcal{S}$ iff the lower sequent of the inference is below $\mathcal{S}$.

5. Let D be a deduction and $\mathcal{S}$ and sequent in D. We call D' the *sub-deduction* of S in D iff D' is a deduction and contains, exactly those sequent, which appears, in every thread of D, above $\mathcal{S}$.

6. A formula φ is called an axiom-formula or end-formula of D iff φ is contained in an axiom or the end-sequent of D.

7. A sequence of formulas is called a *bundle* iff it satisfies the following conditions:

   (a) The sequence begins with an axiom-formula or weakening-formula.

   (b) The sequence ends with an end-formula or cut-formula.

(c) Every formula except the last in the sequence is immediately followed by its successor.

8. Assume φ and ψ are formulas. The formula φ is called the *ancestor* of ψ and ψ is called the *descendent* of φ iff There is a bundle containing φ an ψ in which φ appears before ψ.

9. The notation of implicit and explicit:

   (a) A bundle is called *explicit* iff the the last formula in the bundle is an end-formula.

   (b) A bundle is called *implicit* iff the the last formula in the bundle is an cut-formula.

   (c) A formula in a deduction is called *implicit* or *explicit* iff the bundle which contains the formula is implicit or explicit.

   (d) A sequent in a deduction is called *implicit* or *explicit* iff the sequent contains a formula which is implicit or explicit.

   (e) A logical inference is called *implicit* or *explicit* iff the principal formula of the logical inference is implicit or explicit.

10. A part E of a deduction D is called the *end-piece* of D if it satisfied the following properties:

    (a) The end-sequent is in E.

    (b) The upper sequent of an inference other than an implicit logical inference is contained in E iff the lower sequent of the inference is in E.

    (c) The upper sequent of an implicit logical inference is not in E.

    Or for short: An sequent in a deduction is in the end-piece iff there is no implicit logical inference below this sequent.

11. An inference I is *in the end-piece* of a deduction iff the lower sequent of I is in the end-piece.

12. An inference I is a *boundary* iff the lower sequent of I is in the end-piece and the upper is not.

13. A cut in the end-piece is called *suitable* iff each cut formula of the cut has an ancestor which is the principal formula of a boundary.

Definition 2.38. A deduction D will be called *regular* iff D satisfies:

1. All eigenvariables in D are distinct.

2. If a is a free variable which occurs as a eigenvariable in a sequent $\mathcal{S}$ of D, then a occurs just in sequents above $\mathcal{S}$.

Lemma 2.39. *For every deduction D there is a regular deduction D' with the same end sequent.*

Proof. By a straight forward induction over the application of deduction rules in D where in every step the free variables become replaced such that definition 2.38 is satisfied. □

Lemma 2.40. *If $\boldsymbol{CT}_f \vdash \mathcal{S}(a)$ then for all terms t, $\boldsymbol{CT}_f \vdash \mathcal{S}(t)$.*

Proof. By a straight forward induction over the application of deduction rules in the deduction of $\mathcal{S}(a)$ where in every step a is replaced by t. □

Remark 2.41. In Lemma 2.39 and 2.40 the new deduction is essential the same except that in 2.39 all eigenvariables which occurs twice are replaced by new free variables and in 2.40 every occurrence of a is replaced by t. More precisely, the phrase "essential the same" can be replace in both statements by the notation of a skeleton [3], but since we do not need it again we will not define it here.

The next definition gives a way of counting the complexity of formulas which is very useful in much approaches of proof theory.

Definition 2.42. The *rank* of a formula φ, $\text{rank}(\varphi)$, is defined inductive over definition 2.6:

1. If φ is an atomic formula, then $\text{rank}(\varphi) = 0$.
2. If φ is $\psi \wedge \sigma$, $\psi \vee \sigma$ or $\psi \rightarrow \sigma$, then $\text{rank}(\varphi) = \max\{\text{rank}(\psi), \text{rank}(\sigma)\} + 1$
3. If φ is $\forall x \psi(x)$ or $\exists x \psi(x)$, then $\text{rank}(\varphi) = \text{rank}(\psi(a)) + 1$, where $\psi(x) \equiv \psi(a)[a/x]$

The *rank* of a cut or a IND_f rule is the rank of the cut or induction formula.

Definition 2.43. The *height* of a sequent $\mathcal{S}$ in a deduction D, $h(\mathcal{S}, D)$, is the maximum of the ranks of the cut and IND_f rules in D below $\mathcal{S}$.

From this definition its obvious that.

Proposition 2.44.
1. *If $\mathcal{S}$ is the end-sequent of the deduction D, then $h(\mathcal{S}, D) = 0$.*
2. *If $\mathcal{S}_1$ is above $\mathcal{S}_2$ in a deduction D, then $h(\mathcal{S}_2, D) \leq h(\mathcal{S}_1, D)$.*
3. *If $\mathcal{S}_1$ and $\mathcal{S}_2$ are the upper sequents of the same deduction rule, then $h(\mathcal{S}_1, D) = h(\mathcal{S}_2, D)$.*

The next lemma shows that a equality result based on a identity of two terms is simple.

Lemma 2.45. *If $\boldsymbol{SCT}_f \vdash t = s$ for s, t closed terms, then $\boldsymbol{CT}_f \vdash \varphi(t) \Rightarrow \varphi(s)$ without essential cuts and IND_f.*

Proof. From $\mathbf{SCT}_f \vdash t = s$ follows that there is a $D \vdash t = s$ without any essential cuts and IND_f in $\mathbf{CT}_f$. Now we proof the lemma by induction over the complexity of formulas for some selected cases.

1. If $\varphi(t) \equiv P(t_1, ..., t, ..., t_n)$ then

   $$(1): t_1 = t_1, ..., t = s, ..., t_n = t_n, P(t_1, ..., t, ..., t_n) \Rightarrow P(t_1, ..., s, ..., t_n)$$

   is an axiom. So $P(t_1, ..., t, ..., t_n) \Rightarrow P(t_1, ..., s, ..., t_n)$ can be deducted in $\mathbf{CT}_f$ with (1) and $\mathbf{SCT}_f \vdash\Rightarrow t_i = t_i$ for $1 \leq i \leq n$ with n applications of the cut rule. Because the cut formula of all this cuts is an atom they are inessential.

2. If $\varphi(t) \equiv \psi(t) \wedge \sigma(t)$ then we get from induction hypothesis deductions for

   $$\psi(t) \Rightarrow \psi(s)$$

   $$\sigma(t) \Rightarrow \sigma(s)$$

   without essential cuts and IND_f. With weakening and exchange we get deductions for

   $$\psi(t), \sigma(t) \Rightarrow \psi(s)$$

   $$\psi(t), \sigma(t) \Rightarrow \sigma(s)$$

   with the same property, and so with $\wedge$-right

   $$\psi(t), \sigma(t) \Rightarrow \psi(s) \wedge \sigma(s)$$

   is deducible. Together with two times $\wedge$-left and contraction we can deduct

   $$\psi(t) \wedge \sigma(t) \Rightarrow \psi(s) \wedge \sigma(s).$$

3. If $\varphi(t) \equiv \forall x \psi(x, t)$ then from the induction hypothesis there is a $D \vdash \psi(a, t) \Rightarrow \psi(a, s)$ free from essential cuts and IND_f. Because of Lemma 2.40 we get a $D(t') \vdash \psi(t', t) \Rightarrow \psi(a, s)$ without essential cuts and IND_f for a term t'. From that we get with one application of $\forall$-right and $\forall$-left $\varphi(t) \Rightarrow \varphi(s)$. This finished the proof.

□

2.4 The Consistency of $\mathbf{CT}_f$

At the beginning of a consistency proof of a calculus there should be a definition what it means to be consistent. Following Hilbert, consistency in general means that a calculus is not able to deduct every single syntactical object of the kind the calculus is applied to. In this chapter the syntactical object a calculus is dealing with is the sequent, which leads to the next definition.

Definition 2.46. The calculus $\mathbf{C}$ is *inconsistent* iff $\mathbf{C}$ derives every sequent. Otherwise it is *consistent*.

The next lemma is useful for technical reasons.

Lemma 2.47. *A calculus $\boldsymbol{C}$ is inconsistent iff $\boldsymbol{C} \vdash \Rightarrow$*

Proof. (if): Clear from Definition 2.46.
(only if): Assume $\mathbf{C} \vdash \Rightarrow$ with the deduction D_0 and $\Gamma \Rightarrow \Delta$ is an arbitrary sequent. Since Γ and Δ are finite we can denote there formulas by $\varphi_0, ..., \varphi_n$ and $\psi_0, ..., \psi_m$ where de index follows the appearance in Γ and Δ. By adding weakening left n-times to D starting with φ_n as weakening formula and so on, there is a deduction D_n for $\Gamma \Rightarrow$. After m-times applying weakening right to D_n starting by ψ_0 there is a deduction D_{n+m} such that $D_{n+m} \vdash \Gamma \Rightarrow \Delta$. And from this it follows by definition $\mathbf{C} \vdash \Gamma \Rightarrow \Delta$. □

Remark 2.48. This concept of consistency can be coded into $\mathcal{L}(\tau_{\text{ar}})$ (see [2]) presupposed that $\mathbf{C}$ is defined by a finite set of deduction rules and a primitive recursive set of axioms in a primitive recursive language: Since the set of axioms are primitive recursive and the set of inferences is defined over finite set of deduction rules, which makes it also primitive recursive, the concept of "deduction" can be represented in **PRA** by a 2-array predicate $\mathrm{Prf}_{\mathbf{C}}$,

$$\begin{array}{lll} \mathbf{PRA} \vdash \mathrm{Prf}_{\mathbf{C}}(\ulcorner D \urcorner, \ulcorner \mathcal{S} \urcorner) & \text{iff} & D \text{ is a deduction in } \mathbf{C} \text{ and } D \vdash \mathcal{S} \\ \mathbf{PRA} \vdash \neg\, \mathrm{Prf}_{\mathbf{C}}(\ulcorner D \urcorner, \ulcorner \mathcal{S} \urcorner) & & \text{otherwise} \end{array}$$

where $\mathcal{S}$ is a sequent and $\ulcorner . \urcorner$ is the Gödel number of a finite sequence of symbols.
From this a 1-array predicate is definable via

$$\mathrm{Prf}_{\mathbf{C}}(\bar{n}) \text{ iff } \exists x\, \mathrm{Prf}_{\mathbf{C}}(x, \bar{n}),$$

for $\bar{n} := S^n(0)$, which express in **PRA** if a sequence has a deduction in $\mathbf{C}$. So with the previous lemma we can define

$$\mathrm{Con}(\mathbf{C}) \text{ iff } \neg\, \mathrm{Prf}_{\mathbf{C}}(\ulcorner \Rightarrow \urcorner).$$

From Lemma 2.47 a consistency proof can be given by showing that the empty sequent is not derivable. For syntactical consistency proofs in most cases it is useful to count the length of deductions, such a way to count will be defined next. Note that whenever we talk about ordinals or functions of ordinal arithmetic in this chapter, we really mean ordinal terms of $(\mathbf{OT}_{\varepsilon_0}, \prec_{\varepsilon_0})$ (see Definition 1.46 and Remark 1.47). We will use "$\prec$" for "$\prec_{\varepsilon_0}$".

Definition 2.49. 1. $\omega_0(\alpha) = \alpha$

2. $\omega_{n+1}(\alpha) = \omega^{\omega_n(\alpha)}$

Definition 2.50. Assume D is a deduction in $\mathbf{CT}_f$, then for every sequent $\mathcal{S}$ in D $o(\mathcal{S}, D)$ or $o(\mathcal{S})$ for short is defined inductively:

1. If $\mathcal{S}$ is an axiom in D, then $o(\mathcal{S}) = 1$.

2. If $\mathcal{S}$ is the lower sequent of a weak-structural-rule where $\mathcal{S}_1$ is the upper sequent, then $o(\mathcal{S}) = o(\mathcal{S}_1)$.

3. If $\mathcal{S}$ is the lower sequent of a deduction rule of the form $\wedge$-left, $\vee$-right, $\rightarrow$-right, $\neg$-right, $\neg$-left or of a quantifier rule and $\mathcal{S}_1$ is there upper sequent, then $o(\mathcal{S}) = o(\mathcal{S}_1) + 1$.

4. If $\mathcal{S}$ is the lower sequent of a deduction rule of the form $\wedge$-right, $\vee$-left or $\rightarrow$-left and $\mathcal{S}_1, \mathcal{S}_1$ are there upper sequents, then $o(\mathcal{S}) = o(\mathcal{S}_1) \sharp o(\mathcal{S}_1)$.

5. If $\mathcal{S}$ is the lower sequent of a cut rule and $\mathcal{S}_1, \mathcal{S}_2$ are there upper sequents, then $o(\mathcal{S}) = \omega_{k-l}(o(\mathcal{S}_1) \sharp o(\mathcal{S}_2))$ where k and l are the hight of the upper sequents, respectively.

6. If $\mathcal{S}$ is the lower sequent of a IND_f and $\mathcal{S}_1$ is the upper sequent, then $o(\mathcal{S}) = \omega_{k-l+1}(\mu_1 + 1)$ where μ_1 is the biggest exponent of the normal form of the ordinal $o(\mathcal{S}_1)$ and k, l are the hight of $\mathcal{S}_1$ and $\mathcal{S}$, respectively.

Definition 2.51. The ordinal of a deduction D with $\mathcal{S}$ as end-sequent, in symbols $o(D)$, is defined as $o(D) = o(\mathcal{S}, D)$.

Now we show that the restriction to $(\mathbf{OT}_{\varepsilon_0}, \prec_{\varepsilon_0})$ makes sense.

Proposition 2.52. *(Here, $\prec$ is the full order of $\boldsymbol{OT}$.) For every deduction D in $\boldsymbol{CT}_f$, $o(D) \prec \varepsilon_0$.*

Proof. This follow directly from definition 2.50 and the fact that ε_0 is the first fix point of the function $\lambda\alpha.\omega^\alpha$. $\square$

Lemma 2.53. *For every deduction D in $\boldsymbol{SCT}_f$, $o(D) \prec \omega$.*

Proof. Easy to see because no IND_f and no essential cut is used in $\mathbf{SCT}_f$ which lead to the fact that for every D in $\mathbf{SCT}_f$ and for every $\mathcal{S}$ (if it appears in D) $h(\mathcal{S}, D) = 0$. Since all hights are equal there can never appear a ordinal bigger than ω because of Definition 2.50 Case 5. $\square$

Lemma 2.54. *Assume D is a deduction and $\mathcal{S}$ is a sequent in D without any IND$_f$ below it. Let D_1 be the sub-deduction of $\mathcal{S}$ in D, D_1' any deduction of $\mathcal{S}$ and D' the deduction which is generated by replacing D_1 in D by D_1', then ii follows: If $o(D_1') \prec o(D_1)$, then $o(D') \prec o(D)$.*

Proof. See [16, Ch.2, §12]. □

In generally we can not prove the consistency of $\mathbf{CT}_f$, by using only the consistency of $\mathbf{SCT}_f$ and elementary methods. More exactly it does not hold that $\mathbf{PRA} \vdash \mathrm{Con}(\mathbf{SCT}_f) \rightarrow \mathrm{Con}(\mathbf{CT}_f)$ for every T_f. In particular set $T_f = \mathbf{PA}$. Assume for contradiction that $\mathbf{PRA} \vdash \mathrm{Con}(\mathbf{SCPA}) \rightarrow \mathrm{Con}(\mathbf{CPA})$. Since $\mathbf{PRA} \vdash \mathrm{Con}(\mathbf{SCPA})$ (see [16]) this leads to $\mathbf{PRA} \vdash \mathrm{Con}(\mathbf{CPA})$. But $\mathrm{Con}(\mathbf{CPA})$ is equivalent to $\mathrm{Con}(\mathbf{PA})$ over $\mathbf{PRA}$ (and every proof in $\mathbf{PRA}$ can be done in $\mathbf{PA}$) this leads to $\mathbf{PA} \vdash \mathrm{Con}(\mathbf{PA})$ which contradicts Gödel's second incompleteness theorem.
As Takeuti presents in [16] $\mathbf{PRA} + \mathbf{PRWO}(\varepsilon_0) \vdash \mathrm{Con}(\mathbf{CPA})$, so $\mathbf{PRA}$ can not prove every instance of the schema $\mathbf{PRWO}(\varepsilon_0)$. We present here an adaptation of this proof to prove

$$\mathbf{PRA} + \mathbf{PRWO}(\varepsilon_0) \vdash \mathrm{Con}(\mathbf{SCT}_f) \rightarrow \mathrm{Con}(\mathbf{CT}_f).$$

This proof technique shows, only with primitive recursive methods, that for a deduction D with is a deduction of the empty sequent and not in $\mathbf{SCT}_f$ there is deduction D' which is also an deduction of the empty sequent and $o(D') \prec o(D)$. This leads to a infinite decreasing sequence in $(\mathbf{OT}_{\varepsilon_0}, \prec_{\varepsilon_0})$, which contradicts the statement $\mathbf{PRWO}(\varepsilon_0)$, or to a contradiction in $\mathbf{SCT}_f$, which contradicts $\mathrm{Con}(\mathbf{SCT}_f)$. That $\mathbf{PRWO}(\varepsilon_0)$ is a true arithmetical sentence is obtained by the observations of Chapter 1. Note that Tekeuti gives in [16, p.92-101] a non formal argument for $\mathbf{PRWO}(\varepsilon_0)$ which does not need theories as strong as $\mathbf{ZFC}$. Before the main result is given, a auxiliary lemma must be proven.

Lemma 2.55. *Assume a deduction D in $\mathbf{CT}_f$ satisfy the following properties:*

1. *D is not its own end-piece.*
2. *The end-piece contains neither applications of IND$_f$, weakening nor logical rules.*
3. *If and axiom belong to the end-piece of D it does not contain any logical symbols.*

Then the end-piece of D contains a suitable cut.

Proof. See [16, Ch.2, §12, p.109]. □

Notation 2.56. We use

$$\frac{\mathcal{S}_1 \quad ... \quad \mathcal{S}_n}{\mathcal{S}}$$

to denote that some weak structural rules are applied before the conclusion is inferred.

The next lemma will give the result which leads to the consistency.

Lemma 2.57. *Assume $\boldsymbol{SCT}_f$ is consistent.*
If there is a deductions D in $\boldsymbol{CT}_f$ which $D \vdash \Rightarrow$, then there is a deduction D' in $\boldsymbol{CT}_f$ which $D' \vdash \Rightarrow$ and $o(D') \prec o(D)$.

Proof. Assume there is a deduction D (not in $\mathbf{SCT}_f$) such that $D \vdash \Rightarrow$, from Lemma 2.39 there is a regular deduction $\hat{D} \vdash \Rightarrow$ and $o(D) = o(\hat{D})$ because of the fact that $\hat{D}$ is constructed form D just by renaming variables. The following procedure will construct from $\hat{D}$ a deduction D' which smaller ordinal, it should be obvious that every step can be coded in **PRA** because there are only used primitive recursive operations.

Step 1: Assume $\hat{D}$ contain a free variable a which is not used in $\hat{D}$ as an eigenvariable, then replace a by the constant c_1.

Step 2: Assume the end-piece of $\hat{D}$ contains inferences which are applications of IND$_f$. Assume that $\mathcal{R}$ is the lowest such inference. Suppose $\mathcal{R}$ has the form

$$\frac{\varphi(a_1), ..., \varphi(a_n), \Gamma \Rightarrow \Delta, \varphi(f(a_1, ..., a_n))}{\varphi(c_1), ..., \varphi(c_m), \Gamma \Rightarrow \Delta, \varphi(t)} \mathcal{R}.$$

Let $\mathcal{S}$ be the upper sequent of $\mathcal{R}$ with $h(\mathcal{S}, \hat{D}) = l$, $\mathcal{S}_0$ the lower sequent with $h(\mathcal{S}_0, \hat{D}) = k$ and $\hat{D}_0$ is the subdeduction of $\mathcal{S}$ in $\hat{D}$.
Suppose $o(\mathcal{S}, \hat{D}) = \mu$ with the normal form $\mu = \omega^{\mu_1} + ... + \omega^{\mu_s}$ where $\mu_s \leq ... \leq \mu_1$, then $o(\mathcal{S}_0, \hat{D}) = \omega_{l-k+1}(\mu_1 + 1)$ (2.50).
Because $\mathcal{R}$ is in the end piece of a deduction for the empty sequent there is no logical rule after it. (To deducted the empty sequent all formulas which are introduced anywhere in the deduction have to be cut out. Every logical rule have to be implicit in the deduction. The end piece can not contain any logical rule at all.) So in particular no quantification rule. Together with the facts that $\mathcal{R}$ is the last instance of IND$_f$ in $\hat{D}$ and Step 1 is already applied there is no free variable below $\mathcal{S}$. So the term t in $\mathcal{S}$ is closed. From the assumption of Definition 2.35 $\mathbf{SCT}_f \vdash \Rightarrow t = \bar{t}$ where $\bar{t}$ is a term build up from f and the constants $c_1, ..., c_m$. From Lemma 2.45 follows that $\mathbf{CT}_f \vdash \varphi(\bar{t}) \Rightarrow \varphi(t)$ via P without essential cuts and IND$_f$ and so $o(\varphi(\bar{t}) \Rightarrow \varphi(t), P) = p \prec \omega$.
Via induction over complexity of the term $\bar{t}$ we proof that

$$\mathcal{S}_0' : \varphi(c_1), ..., \varphi(c_m), \Gamma \Rightarrow \Delta, \varphi(\bar{t})$$

have a deduction $D'_{0.0}$ such that together with the deduction P we get a deduction D'_0 of $\mathcal{S}$ via the application of the cut rule

$$\frac{\varphi(c_1),...,\varphi(c_m),\Gamma \Rightarrow \Delta,\varphi(\bar{t}) \quad \varphi(\bar{t}) \Rightarrow \varphi(t)}{\varphi(c_1),...,\varphi(c_m),\Gamma \Rightarrow \Delta,\varphi(t)}.$$

From D'_0, D' can be constructed by replacing $\hat{D}_0$ with D'_0, such that $o(\mathcal{S}'_0, D') \prec q * \mu$ for some q (here $q * \mu$ means $\mu\sharp...\sharp\mu$, q many times):
Induction start: Assume $\bar{t} \equiv c_i$ for $1 \leq i \leq m$. The sequent $\varphi(c_i) \Rightarrow \varphi(c_i)$ can be deduct without applications of cut and IND_f. From this sequent we get with finite instances of the weakening rule a deduction $D'_{0.0}$ for $\varphi(c_1),...,\varphi(c_m),\Gamma \Rightarrow \Delta,\varphi(c_i)$ such that

$$o(\varphi(c_1),...,\varphi(c_m),\Gamma \Rightarrow \Delta,\varphi(c_i), D') = q \prec q * \mu.$$

Induction step: Assume $\bar{t} \equiv f(t_1,...,t_n)$ where $t_1,...,t_n$ are terms build up just by f and $c_1,...,c_m$. From induction hypothesis follows that there are deductions $D'_{0,i}$ for

$$\mathcal{S}_i : \varphi(c_1),...,\varphi(c_m),\Gamma \Rightarrow \Delta,\varphi(t_i)$$

with $1 \leq i \leq n$ and $o(\mathcal{S}_i, D') \prec q_i * \mu$.
From $\hat{D}_0$ we get a deduction for

$$\mathcal{S}' : \varphi(t_1),...,\varphi(t_n),\Gamma \Rightarrow \Delta,\varphi(f(t_1,...,t_n))$$

with Lemma 2.40 such that $\hat{D}_0(t_1,...,t_n)$ is constructed by substitute t_i for a_i in $\hat{D}_0$ (which do not makes a different for the ordinal). Then we use cuts of the form:

$$\frac{\varphi(c_1),...,\varphi(c_m),\Gamma \Rightarrow \Delta,\varphi(t_1) \quad \varphi(t_1),...,\varphi(t_n),\Gamma \Rightarrow \Delta,\varphi(f(t_1,...,t_n))}{\varphi(c_1),\varphi(c_2),...,\varphi(c_m),\varphi(t_2),...,\varphi(t_n),\Gamma \Rightarrow \Delta,\varphi(f(t_1,...,t_n))}\ \mathcal{C}_1$$

where we denote the lowest sequent by $\mathcal{S}_{0,1}$. And also cuts together with some weak structural rule to get this figure:

$$\frac{\varphi(c_1),...,\varphi(c_m),\Gamma \Rightarrow \Delta,\varphi(t_1) \quad \mathcal{C}_1}{\varphi(c_1),\varphi(c_2),...,\varphi(c_m),\varphi(t_3),...,\varphi(t_n),\Gamma \Rightarrow \Delta,\varphi(f(t_1,...,t_n))}\ \mathcal{C}_2$$

where again $\mathcal{S}_{0,2}$ denotes the lowest sequent. So after n of the $\mathcal{C}_j \vdash \mathcal{S}_{0,j}$ a deduction $D'_{0.0}$ for

$$\varphi(c_1),\varphi(c_2),...,\varphi(c_m),\Gamma \Rightarrow \Delta,\varphi(f(t_1,...,t_n))$$

can be obtained. Since $h(\mathcal{S}_j,\hat{D}) = h(\mathcal{S}_j, D') = r$ for every $1 \leq j \leq n$, because the cut formulas of the $\mathcal{C}_j$ have all the same rank and so they are all affected in the same amount from the cuts and IND_f below $\mathcal{S}$, it follows

$o(\mathcal{S}_j, D') \prec \sharp_{i=1}^{j} q_i * \mu$, and since $o(\mathcal{S}', D') = o(\mathcal{S}, \hat{D}) = \mu$, this leads to $o(\mathcal{S}_0', D') \prec \sharp_{i=1}^{n} q_i * \mu$ which finished the induction.
As explained above D_0' is build up form $P \vdash \varphi(\bar{t}) \Rightarrow \varphi(t)$ $(o(P) = p \prec \omega)$ and the deduction $D_{0,0}' \vdash \mathcal{S}_0'$ to deduct via

$$\frac{\varphi(c_1), ..., \varphi(c_m), \Gamma \Rightarrow \Delta, \varphi(\bar{t}) \quad \varphi(\bar{t}) \Rightarrow \varphi(t)}{\varphi(c_1), ..., \varphi(c_m), \Gamma \Rightarrow \Delta, \varphi(t)}$$

the sequent $\mathcal{S}$. Replace in $\hat{D}$ the supdeduction $\hat{D}_0$ by this new deduction to get the deduction D' from $\hat{D}$, as also explained above. Then for some $q \in \omega$,

$$o(\mathcal{S}, D') = \omega_{l-k}(q * \mu + p) \prec \omega_{l-k+1}(\mu_1 + 1) = o(\mathcal{S}, \hat{D}),$$

because $q * \mu + p \prec \omega^{\mu_1 + 1}$ since $p \prec \omega$. Which lead with Lemma 2.54 to

$$o(D') \prec o(\hat{D}) = o(D).$$

So we assume there is no IND_f in the end-piece of $\hat{D}$.

Step 3: Assume the end-piece of $\hat{D}$ contains a logical axiom, say $\varphi \Rightarrow \varphi$. Because $\hat{D} \vdash \Rightarrow$, φ have to be cut out on both sides. It have to be φ because we are in the end-piece of a deduction where no logical rules are appear, so all descendants of φ are φ itself. We confirm the case where φ from the antecedence is cut out first, the case where φ from the succedence is cut out first is analogous. Consider the supdeduction $\hat{D}_0$ which end with the cut

$$\frac{\Gamma \Rightarrow \Delta, \varphi \quad \varphi, \Pi \Rightarrow \Lambda}{\Gamma, \Pi \Rightarrow \Delta, \Lambda}$$

where φ (from the succedence) is in Λ. So we can deduct $\Gamma, \Pi \Rightarrow \Delta, \Lambda$ (call it $\mathcal{S}$) from $\Gamma \Rightarrow \Delta, \varphi$ just by use of weakening and exchange and get a new deduction for $\mathcal{S}$ called D_0'. By replacing $\hat{D}_0$ with D_0' in $\hat{D}$ we get a deduction D'. Since weakening and exchange do not count in the ordinal countering

$$o(\mathcal{S}, D') \prec o(\mathcal{S}, \hat{D}).$$

From Lemma 2.54 it follows that

$$o(D') \prec o(\hat{D}) = o(D).$$

So we assume there is no logical axiom in the end-piece of $\hat{D}$.

Step 4: Assume there is a weakening in the end-piece and let $\mathcal{R}$ be the lowest of these. Since $\hat{D} \vdash \Rightarrow$ there have to be a cut $\mathcal{C}$ below $\mathcal{R}$ such that the principal formula of $\mathcal{R}$ is the cut formula of $\mathcal{C}$ (because the descendent of a formula have to be the formula itself in the end-piece). So a part of $\hat{D}$ have the form:

$$\dfrac{\dfrac{}{\Gamma \Rightarrow \Delta, \varphi} \quad \dfrac{\dfrac{\Pi' \Rightarrow \Lambda'}{\varphi, \Pi' \Rightarrow \Lambda'}\,\mathcal{R}}{\begin{array}{c}\vdots\\ \varphi, \Pi \Rightarrow \Lambda\end{array}}}{\Gamma, \Pi \Rightarrow \Delta, \Lambda}\,\mathcal{C} \quad .$$

Case 1: If no contraction is applied to φ between $\mathcal{R}$ and $\mathcal{C}$, reduce $\hat{D}$ to D' by replacing the considered part of $\hat{D}$ by the following:

$$\dfrac{\begin{array}{c}\Pi' \Rightarrow \Lambda' \\ \vdots \\ \Pi \Rightarrow \Lambda\end{array}}{\Gamma, \Pi \Rightarrow \Delta, \Lambda}\;\text{weakenings and exchanges} \quad .$$

Assume $h(\Gamma, \Pi \Rightarrow \Delta, \Lambda, \hat{D}) = l$ and $h(\varphi, \Pi \Rightarrow \Lambda, \hat{D}) = k$. Then, $l \preceq k$ and $h(\Pi \Rightarrow \Lambda, D') = h(\Gamma, \Pi \Rightarrow \Delta, \Lambda, D') = l$. Let $\mathcal{S}$ a sequent in $\hat{D}$ above $\varphi, \Pi \Rightarrow \Lambda$ and let $\mathcal{S}'$ be the corresponding sequent in D'. Then by induction on the number of applications of rules up to $\varphi, \Pi \Rightarrow \Lambda$ it is possible to show

$$\omega_{k_1 - k_2}(o(\mathcal{S}, \hat{D})) \succeq o(\mathcal{S}', D')$$

where $k_1 = h(\mathcal{S}, \hat{D})$ and $k_2 = h(\mathcal{S}', D')$. Hence, if

$$\begin{aligned} \mu_1 &= o(\Gamma \Rightarrow \Delta, \varphi, \hat{D}), & \mu_2 &= o(\varphi, \Pi \Rightarrow \Lambda, \hat{D}), \\ \nu &= o(\Gamma, \Pi \Rightarrow \Delta, \Lambda, \hat{D}), & \mu_2' &= o(\Pi \Rightarrow \Lambda, \hat{D}), \\ \nu' &= o(\Gamma, \Pi \Rightarrow \Delta, \Lambda, D'), \end{aligned}$$

then

$$\omega_{k-l}(\mu_2) \succeq \mu_2'$$

and further,

$$\nu = \omega_{k-l}(\mu_2 \sharp \mu_1) \succ \omega_{k-l}(\mu_2) \succeq \mu_2' = \nu'.$$

So by Lemma 2.54 it follows that

$$o(D') \prec o(\hat{D}) = o(D).$$

Case 2: Assume not case 1, then we consider the uppermost application of contraction between $\mathcal{R}$ and $\mathcal{C}$ and replaced it as described below to get a deduction E from $\hat{D}$.

$$
\begin{array}{cc}
\hat{D} & E \\
\dfrac{\Pi' \Rightarrow \Lambda'}{\varphi, \Pi' \Rightarrow \Lambda'} & \Pi' \Rightarrow \Lambda' \\
\vdots & \vdots \\
\dfrac{\varphi, \varphi, \Pi'' \Rightarrow \Lambda''}{\varphi, \Pi'' \Rightarrow \Lambda''} & \varphi, \Pi'' \Rightarrow \Lambda'' \\
\vdots & \vdots \\
\varphi, \Pi \Rightarrow \Lambda & \varphi, \Pi \Rightarrow \Lambda
\end{array}
$$

Clearly $o(E) = o(\hat{D}) = o(D)$.

From now on assume there is no application of weakening in the end-piece of $\hat{D}$.

Step 5: $\hat{D} \vdash \Rightarrow$ can not be his own end-piece. Otherwise $\hat{D}$ would be simple which is impossible by assumption. By Lemma 2.55 there is a suitable cut in the end-piece of $\hat{D}$. The lowermost of this cuts, called $\mathcal{C}$, will be reduced.
Case 1: The cut formula of $\mathcal{C}$ have the form $\phi \wedge \psi$, so $\hat{D}$ have the form

$$
\begin{array}{c}
\dfrac{
\begin{array}{c}
\dfrac{\begin{array}{c}\vdots \\ \Gamma' \Rightarrow \Theta', \varphi\end{array} \quad \begin{array}{c}\vdots \\ \Gamma' \Rightarrow \Theta', \psi\end{array}}{\Gamma' \Rightarrow \Theta', \varphi \wedge \psi}\,\mathcal{R}_1 \\
\vdots \\
\Gamma \Rightarrow \Theta, \varphi \wedge \psi
\end{array}
\qquad
\begin{array}{c}
\dfrac{\begin{array}{c}\vdots \\ \varphi, \Pi' \Rightarrow \Lambda'\end{array}}{\varphi \wedge \psi, \Pi' \Rightarrow \Lambda'}\,\mathcal{R}_2 \\
\vdots \\
\varphi \wedge \psi, \Pi \Rightarrow \Lambda
\end{array}
}{\Gamma, \Pi \Rightarrow \Theta, \Lambda}\,\mathcal{C} \\
\vdots \\
\Delta \Rightarrow \Xi \\
\vdots \\
\Rightarrow
\end{array}
$$

where $\Delta \Rightarrow \Xi$ denotes the first sequent such that

$$
k = h(\Delta \Rightarrow \Xi, \hat{D}) \prec h(\Gamma \Rightarrow \Theta, \varphi \wedge \psi, \hat{D}) = h(\varphi \wedge \psi, \Pi \Rightarrow \Lambda, \hat{D}) = l.
$$

The existence of this sequent is ensured by Proposition 2.44. Note, $\Delta \Rightarrow \Xi$ can be the lower sequent of $\mathcal{C}$ or the end-sequent.
Since $k \prec l$ and there is no IND_f in the end-piece, $\Delta \Rightarrow \Xi$ must be the lower sequent of a cut, called $\mathcal{R}_3$. Assume

$$
\begin{aligned}
\mu =& o(\Gamma \Rightarrow \Theta, \phi \wedge \psi, \hat{D}) \\
\nu =& o(\varphi \wedge \psi, \Pi \Rightarrow \Lambda, \hat{D}) \\
\lambda =& o(\Delta \Rightarrow \Xi, \hat{D}).
\end{aligned}
$$

Consider the following deductions:
D'_1:

$$\begin{array}{c}
\dfrac{\begin{array}{c}\vdots\\ \dfrac{\dfrac{\Gamma' \Rightarrow \Theta', \varphi}{\Gamma' \Rightarrow \varphi, \Theta'}}{\Gamma' \Rightarrow \varphi, \Theta', \varphi \wedge \psi}\ \text{weakening}\\ \vdots\\ \Gamma \Rightarrow \varphi, \Theta, \varphi \wedge \psi\end{array} \qquad \begin{array}{c}\vdots\\ \varphi \wedge \psi, \Pi \Rightarrow \Lambda\end{array}}{\Gamma, \Pi \Rightarrow \varphi, \Theta, \Lambda}\ \mathcal{R}_{3_1}\\
\vdots\\
\dfrac{\Delta \Rightarrow \varphi, \Xi}{\Delta \Rightarrow \Xi, \varphi}
\end{array}$$

D'_2:

$$\begin{array}{c}
\dfrac{\begin{array}{c}\vdots\\ \Gamma \Rightarrow \Theta, \varphi \wedge \psi\end{array} \qquad \begin{array}{c}\vdots\\ \dfrac{\dfrac{\varphi, \Pi' \Rightarrow \Lambda'}{\Pi', \varphi \Rightarrow \Lambda'}}{\varphi \wedge \psi, \Pi', \varphi \Rightarrow \Lambda'}\ \text{weakening}\\ \vdots\\ \varphi \wedge \psi, \Pi, \varphi \Rightarrow \Lambda\end{array}}{\Gamma, \Pi, \varphi \Rightarrow \Theta, \Lambda}\ \mathcal{R}_{3_2}\\
\vdots\\
\dfrac{\Delta, \varphi \Rightarrow \Xi}{\varphi, \Delta \Rightarrow \Xi}
\end{array}$$

From that D' is constructed form D'_1 and D'_2 via sticking both together with a cut and follows from $\Delta \Rightarrow \Xi$ like in $\hat{D}$.

$$\begin{array}{c}
\dfrac{\dfrac{\dfrac{D'_1}{\Delta \Rightarrow \Xi, \varphi} \quad \dfrac{D'_2}{\varphi, \Delta \Rightarrow \Xi}}{\Delta, \Delta \Rightarrow \Xi, \Xi}}{\Delta \Rightarrow \Xi}\\
\vdots\\
\Rightarrow
\end{array}$$

Assume that $h(\Delta \Rightarrow \Xi, \varphi, D') = h(\varphi, \Delta \Rightarrow \Xi, D') = m$ and note that

$$h(\Delta, \Delta \Rightarrow \Xi, \Xi, D') = h(\Delta \Rightarrow \Xi, D') = h(\Delta \Rightarrow \Xi, \hat{D}) = k.$$

It is obvious that

$$m = \begin{cases} k & : \ k \succ \operatorname{rank}(\varphi) \\ \operatorname{rank}(\varphi) & : \ \text{otherwise.} \end{cases}$$

In both cases $k \preceq m \prec l$. It follows

$$h(\Gamma \Rightarrow \varphi, \Theta, \varphi \wedge \psi, D') = h(\varphi \wedge \psi, \Pi \Rightarrow \Lambda, D') = l,$$

because all cut formulas below $\mathcal{C}$ in $\hat{D}$ occur in D' below $\mathcal{R}_{3_1}$, all cut formulas below $\mathcal{R}_{3_1}$ in D' except φ occurs in $\hat{D}$ under $\mathcal{C}$ and it holds that $\mathrm{rank}(\varphi) \prec \mathrm{rank}(\varphi \wedge \psi) \preceq l$. And a similar argumentation for

$$h(\Gamma \Rightarrow \Theta, \varphi \wedge \psi, D') = h(\varphi \wedge \psi, \Pi, \varphi \Rightarrow \Lambda, D') = l.$$

Assume

$$\begin{aligned} \mu_1 &= o(\Gamma \Rightarrow \varphi, \Theta, \phi \wedge \psi, D'), & \nu_1 &= o(\varphi \wedge \psi, \Pi \Rightarrow \Lambda, D'), \\ \lambda_1 &= o(\Delta \Rightarrow \varphi, \Xi, D'), & \mu_2 &= o(\Gamma \Rightarrow \Theta, \varphi \wedge \psi, D'), \\ \nu_2 &= o(\varphi \wedge \psi, \Pi, \varphi \Rightarrow \Lambda, D'), & \lambda_2 &= o(\Delta, \varphi \Rightarrow \Xi, D'), \\ \lambda_0 &= o(\Delta, \Delta \Rightarrow \Xi, \Xi, D'). \end{aligned}$$

Then $\mu_1 \prec \mu$, $\nu_1 = \nu$, $\mu_2 = \mu$ and $\nu_2 \prec \nu$.
Assume

$$\frac{\mathcal{S}'_1 \quad \mathcal{S}'_2}{\mathcal{S}'}\ \mathcal{R}'$$

is an arbitrary deduction rule between $\mathcal{R}_{3_1}$ and $\Delta \Rightarrow \varphi, \Xi$ in D' and assume

$$\frac{\mathcal{S}_1 \quad \mathcal{S}_2}{\mathcal{S}}\ \mathcal{R}$$

is the corresponding deduction rule between $\mathcal{C}$ and $\Delta \Rightarrow \Xi$ in $\hat{D}$. Assume

$$\begin{aligned} \alpha'_1 &= o(\mathcal{S}'_1, D'), & \alpha'_2 &= o(\mathcal{S}'_2, D'), & \alpha' &= o(\mathcal{S}', D'), \\ \alpha_1 &= o(\mathcal{S}_1, D'), & \alpha_2 &= o(\mathcal{S}_2, \hat{D}), & \alpha &= o(\mathcal{S}, \hat{D}), \end{aligned}$$

$$\begin{aligned} h(\mathcal{S}'_1, D') = h(\mathcal{S}'_2, D') &= k_1, \\ h(\mathcal{S}', D') &= k_2. \end{aligned}$$

Then

$$\alpha = \begin{cases} \alpha_1 \sharp \alpha_2 & : \ \mathcal{S} \not\equiv \Delta \Rightarrow \varphi, \Xi \\ \omega_{l-k}(\alpha_1 \sharp \alpha_2) & : \ \mathcal{S} \equiv \Delta \Rightarrow \varphi, \Xi \end{cases}$$

and $\alpha' = \omega_{k_1 - k_2}(\alpha'_1 \sharp \alpha'_2)$. By and induction, starting with $\mu_1 \prec \mu$ and $\nu_1 = \nu$, on the number of applications of deduction rules between $\mathcal{R}_{3_1}$ and $\mathcal{S}$ it is possible to show that,

$$\text{if } \mathcal{S} \not\equiv \Delta \Rightarrow \varphi, \Xi, \text{ then } \alpha' \prec \omega_{l-k_2}(\alpha). \quad (\star)$$

Assume $\lambda = \omega_{l-k}(\kappa)$. From $(\star)$ it follows $\lambda_1 \prec \omega_{l-m}(\kappa)$. A similar argumentation shows that $\lambda_2 \prec \omega_{l-m}(\kappa)$.
From $l - k = (l - m) + (m - k)$ follows,

$$\omega_{m-k}(\lambda_1 \sharp \lambda_2) \prec \omega_{l-k}(\kappa).$$

This implies $\lambda_0 \prec \lambda$, which lead as in the steps before with Lemma 2.54 to

$$o(D') \prec o(\hat{D}) = o(D).$$

Case 2: The cut formula of $\mathcal{C}$ have the form $\forall x\varphi(x)$, so $\hat{D}$ have the form

$$\begin{prooftree}
\AxiomC{$\vdots$}
\noLine
\UnaryInfC{$\Gamma' \Rightarrow \Theta', \varphi(a)$}
\RightLabel{$\mathcal{R}_1$}
\UnaryInfC{$\Gamma' \Rightarrow \Theta', \forall x\varphi(x)$}
\noLine
\UnaryInfC{$\vdots$}
\noLine
\UnaryInfC{$\Gamma \Rightarrow \Theta, \forall x\varphi(x)$}
\AxiomC{$\vdots$}
\noLine
\UnaryInfC{$\varphi(s), \Pi' \Rightarrow \Lambda'$}
\RightLabel{$\mathcal{R}_2$}
\UnaryInfC{$\forall x\varphi(x), \Pi' \Rightarrow \Lambda'$}
\noLine
\UnaryInfC{$\vdots$}
\noLine
\UnaryInfC{$\forall x\varphi(x), \Pi \Rightarrow \Lambda$}
\RightLabel{$\mathcal{C}$}
\BinaryInfC{$\Gamma, \Pi \Rightarrow \Theta, \Lambda$}
\noLine
\UnaryInfC{$\vdots$}
\noLine
\UnaryInfC{$\Delta \Rightarrow \Xi$}
\noLine
\UnaryInfC{$\vdots$}
\noLine
\UnaryInfC{$\Rightarrow$}
\end{prooftree}$$

where $\Delta \Rightarrow \Xi$ is defined as in case 1 and D' is constructed from two deductions D'_1 and D'_2 defined as follows.
D'_1 :

$$\begin{prooftree}
\AxiomC{$\vdots$}
\noLine
\UnaryInfC{$\Gamma' \Rightarrow \Theta', \varphi(s)$}
\doubleLine
\UnaryInfC{$\Gamma' \Rightarrow \varphi(s), \Theta'$}
\RightLabel{weakening}
\UnaryInfC{$\Gamma' \Rightarrow \varphi(s), \Theta', \forall x\varphi(x)$}
\noLine
\UnaryInfC{$\vdots$}
\noLine
\UnaryInfC{$\Gamma \Rightarrow \varphi(s), \Theta, \forall x\varphi(x)$}
\AxiomC{$\vdots$}
\noLine
\UnaryInfC{$\forall x\varphi(x), \Pi \Rightarrow \Lambda$}
\BinaryInfC{$\Gamma, \Pi \Rightarrow \varphi(s), \Theta, \Lambda$}
\noLine
\UnaryInfC{$\vdots$}
\noLine
\UnaryInfC{$\Delta \Rightarrow \varphi(s), \Xi$}
\doubleLine
\UnaryInfC{$\Delta \Rightarrow \Xi, \varphi(s)$}
\end{prooftree}$$

where the deduction of $\Gamma' \Rightarrow \Theta', \varphi(s)$ comes from the deduction of
$\Gamma' \Rightarrow \Theta', \varphi(a)$ via Lemma 2.40.

D'_2 :

$$
\begin{prooftree}
\AxiomC{$\vdots$}
\noLine
\UnaryInfC{$\varphi(s), \Pi' \Rightarrow \Lambda'$}
\doubleLine
\UnaryInfC{$\Pi', \varphi(s) \Rightarrow \Lambda'$}
\RightLabel{weakening}
\UnaryInfC{$\forall x\varphi(x), \Pi', \varphi(s) \Rightarrow \Lambda'$}
\noLine
\UnaryInfC{$\vdots$}
\noLine
\UnaryInfC{$\forall x\varphi(x), \Pi, \varphi(s) \Rightarrow \Lambda$}
\AxiomC{$\vdots$}
\noLine
\UnaryInfC{$\Gamma \Rightarrow \Theta, \forall x\varphi(x)$}
\BinaryInfC{$\Gamma, \Pi, \varphi(s) \Rightarrow \Theta, \Lambda$}
\noLine
\UnaryInfC{$\vdots$}
\noLine
\UnaryInfC{$\Delta, \varphi(s) \Rightarrow \Xi$}
\doubleLine
\UnaryInfC{$\varphi(s), \Delta \Rightarrow \Xi$}
\end{prooftree}
$$

From this two deductions, D' is defined as follows:

$$
\begin{prooftree}
\AxiomC{D'_1}
\UnaryInfC{$\Delta \Rightarrow \Xi, \varphi(s)$}
\AxiomC{D'_2}
\UnaryInfC{$\varphi(s), \Delta \Rightarrow \Xi$}
\BinaryInfC{$\Delta, \Delta \Rightarrow \Xi, \Xi$}
\doubleLine
\UnaryInfC{$\Delta \Rightarrow \Xi$}
\noLine
\UnaryInfC{$\vdots$}
\noLine
\UnaryInfC{$\Rightarrow$}
\end{prooftree}
$$

Since $o(\Gamma' \Rightarrow \Theta', \varphi(s)) = o(\Gamma' \Rightarrow \Theta', \varphi(a))$ the rest of the proof is essentially the same as in Case 1.

For the rest of the cases the proof is very similar to the two which are shown. This finished the proof. □

Theorem 2.58. ***Relative Consistency of $\mathbf{CT}_f$***
If $\mathbf{SCT}_f$ is consistent, then also $\mathbf{CT}_f$.

Proof. Assume $\mathbf{SCT}_f$ is consistent and $\mathbf{CT}_f$ is inconsistent. Then there is a deduction D in $\mathbf{CT}_f$ such that $D \vdash \Rightarrow$. From the consistency of $\mathbf{SCT}_f$ together with Lemma 2.57 follows that there is infinite decreasing sequence in $(\mathrm{OT}_{\varepsilon_0}, \prec_{\varepsilon_0})$. This is a contradiction to the fact that $(\mathrm{OT}_{\varepsilon_0}, \prec_{\varepsilon_0})$ is well-founded. □

By the method descried above one can prove the next corollary.

Corollary 2.59. *Assume that $\varphi(a)$ is quantifier free and that T_f is consistent.*
If $T_f \models \exists x\varphi(x)$ then $(T_f)_0 \models \exists x\varphi(x)$.
I.e., T_f is Σ^0_1-conservative over $(T_f)_0$.

Proof. Assume that $T_f \models \exists x\varphi(x)$ where $\varphi(a)$ is quantifier free. Then $T'_f := T_f \cup \{\forall x \neg\varphi(x)\}$ is inconsistent and satisfies Assumption 2.33 (since T_f does).

So $\mathbf{CT}'_f \vdash \Rightarrow$ by Remark 2.36. By Lemma 2.58 this implies $\mathbf{SCT}'_f$ is inconsistent. According to Lemma 2.32 this implies that $(T'_f)_0 = (T_f)_0 \cup \{\forall x \neg\varphi(x)\}$ is inconsistent. So we obtain $(T_f)_0 \models \exists x\varphi(x)$. ☐

The next corollary shows that **PRA** is sufficient as basic theory.

Corollary 2.60. *The theory* $\boldsymbol{PRA}$ *plus* $\boldsymbol{PRWO}(\varepsilon_0)$ *proves:*
If $\mathrm{Con}(\boldsymbol{SCT}_f)$ *and " For all closed terms t there is an f-term $\bar{t}$ s.t.* $\boldsymbol{SCT}_f \vdash t = \bar{t}$.*", then* $\mathrm{Con}(\boldsymbol{CT}_f)$.

Proof. The argument in the proof of 2.58 can be carried out in

$$\mathbf{PRA} + \mathbf{PRWO}(\varepsilon_0)$$

by a standard coding because all operations in Lemma 2.57 are primitive recursive. ☐

2.5 Examples for T_f

We end this chapter by a collection of examples of the considered theories T_f.

Example 2.61. Let $\tau = \{0, S, +, \times\}$ (called τ_{ar}). We denote the theory which contains the induction axioms for S and 0 together with the formulas

$$\forall x \neg(S(x) = 0)$$

$$\forall x \forall y (S(x) = S(y) \rightarrow x = y)$$

$$\forall x (x + 0 = x)$$

$$\forall x \forall y (x + S(y) = S(x + y))$$

$$\forall x (x \times 0 = 0)$$

$$\forall x \forall y (x \times S(y) = x \times y + x)$$

by **PA**, called *peano arithmetic*. It is a well known fact (see [16]) that **PA** satisfies Assumption 2.33. Here **SCPA** is only the equational part of peano arithmetic between two closed terms build up from $\{0, S, +, \times\}$. **SCPA** is weaker than Robinsons Q (in the sense that SCPA can not prove that every natural number have a predecessor, also $\mathbb{Q}$ is a model of **SCPA** by the standard interpretation of the constance). This is the classical example for Gentzen's proof.

Example 2.62. A Group theory $G_{a,b}$, formulated in the language $\tau = \{e, a, b, \circ, ^{-1}\}$, as

$$\forall x \forall y \forall z ((x \circ y) \circ z = x \circ (y \circ z))$$

$$\forall x (x \circ e = x)$$

$$\forall x(x \circ x^{-1} = e)$$
$$e \neq a$$
$$e \neq b$$
$$a \neq b$$
$$(a \circ b^{-1}) \circ a = a \circ (b^{-1} \circ a)$$

(where we denote $\circ(x, y)$ by $x \circ y$) together with an induction schema over $a, b, \circ$ and $^{-1}$. **SCG** will be as above the equational part of closed terms. But considering consistency alone, this example is rather pointless since $G_{a,b}$ has a finite model (e.g. $\mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}$).

Example 2.63. Since **ZFC** is a primitive recursive axiom system the sentence $\mathrm{Con}(\mathbf{ZFC})$ is a universal sentence with primitive recursive matrix. This makes $\mathbf{PA} + \mathrm{Con}(\mathbf{ZFC})$ to an example. Because the matrix of $\mathrm{Con}(\mathbf{ZFC})$ is primitive recursive the simple part in this example is as strong as $\mathrm{Con}(\mathbf{ZFC})$. So of course $\mathbf{PRA}+\mathbf{PRWO}(\varepsilon_0)$ can not prove $\mathrm{Con}(T_f)$.

Example 2.64. Let

$$f(x) := \begin{cases} x+1 & : \quad \neg[(\exists y \leq x)\neg \mathrm{Prf}_{\mathbf{ZFC}}(y, \perp)] \\ 0 & : \quad \text{otherwise,} \end{cases}$$

which is obviously primitive recursive. Define T_f as $\mathbf{PRA}_0$ plus all induction axioms for f and 0, then $T_f \models \mathrm{Con}(\mathbf{ZFC})$. (Since $\neg\,\mathrm{Con}(\mathbf{ZFC})$ implies that $f(x)$ is bounded by some N. So by induction over f we could prove $\forall x(x < N)$, which is obviously inconsistent.) The theory T_f also has the property demanded in Assumption 2.33, because $f(x) = S(x)$ by assuming $\mathrm{Con}(\mathbf{ZFC})$. Since one have to assume $\mathrm{Con}(\mathbf{ZFC})$ the theory **PRA** is not strong enough to prove that $\mathbf{SCT}_f$ satisfy Assumption 2.33. (Note Corollary 2.60 in this context.)

3 The Π^1_1-Ordinal from Cut-Elimination

As it will be explain later, the following method, introduced by Lorenzen, Schütte, Tait and others, needs stronger restrictions on the considered theories: Basically, all function and predicate constants have to have a standard interpretation in $\mathbb{N}$ and the theory may only prove true sentences (in the standard interpretation).

In this chapter we consider theories which again include the generalised induction axioms (as in Chapter 2) for a function f and individual constants $c_1, ..., c_m$ and all defining axioms for primitive recursive functions.

We are not interested in consistency questions as in the notion of proof theoretical ordinal as defined in Definition 1.48 in this chapter. Because we are interested to get an upper bound for the Π^1_1-Ordinal (see Definition 3.51) of the considered Theories from a cut-elimination procedure, we have to enlarge the language by (free) second order variables. However, similar methods can be used to find the proof theoretical ordinal of Definition 1.48 of the theories without second order variables, see [4] and [13].

This chapter is closely related to [15] and [8]. We will proceed as follows:

1. We introduce a finite Tait language where free second order variables are included to get (pseudo) Π^1_1-sentences.
2. A finite Tait style calculus TA_f of the considered theories is given.
3. We give an infinite Tait-calculus NT_∞ which has cut elimination (via primitive recursive operations on infinite trees).
4. To any deduction in NT_∞ we assign an ordinal (not an ordinal term!) called the "rank" of the deduction.
5. The crucial point will be to translate TA_f-deductions into NT_∞-deductions. The translated deductions are bounded by ω^2.
6. By repeating the cut elimination procedure we can show that the cut free versions of the translated deductions have at most rank ε_0.
7. We give a staged version of the truth definition for sentences ($\models^\alpha$).
8. If φ has a cut-free deduction of rank α in NT_∞, then $\mathbb{N} \models^\alpha \varphi$.
9. We introduce the notation of the Π^1_1-ordinal of a theory T, $\|T\|_{\Pi^1_1}$. This will be the smallest $\alpha \in \mathbf{ON}$ such that a T-provable Π^1_1-formula φ satisfies $\mathbb{N} \models^\alpha \varphi$.
10. If a Π^1_1-formula φ has a deduction in the finite calculus by translation it has a deduction in the infinite calculus bounded by ω^2. So it has a cut free (infinite) deduction bounded by ε_0. This implys $\mathbb{N} \models^{\varepsilon_0} \varphi$. So we get $\|T\|_{\Pi^1_1} \leq \varepsilon_0$.

3.1 The Language of a Tait Calculus

To introduce a (finite) Tait calculus TA_f we have to remove negation from the connective and define it on the atomic level instead. In this section we give a finite language, so disjunction and conjunction have only finite many components. Note that we will define a infinite language later in Section 3.3. We also have second order variables in the language but not second order quantification.

Definition 3.1. The set of *primitive symbols* is the following:

1. Logical Symbols:
   (a) Logical connectives: $\wedge, \vee$
   (b) Quantifiers: $\exists, \forall$
   (c) First order free variables: $a_0, a_1, a_2, ...$
   (d) First order bound variables: $x_0, x_1, x_2, ...$
   (e) Second order variables: $X_0, X_1, X_2, ...$
   (f) Relationship symbols: $\in, \notin$
   (g) Brackets: $($, $)$
2. Constants: The set τ_{nt} of constants consists of:
   (a) An individual constant for each natural number: $\bar{0}, \bar{1}, \bar{2}, \bar{3}, ...$
   (b) The 1-array function constant S "successor" and for every other n-array primitive recursive function a n-array function constant: $f_0^n, f_1^n, f_2^n, ...$
   (c) Two 2-array predicate constants $=$ and $\neq$. And for every primitive recursive n-array relation a predicate constant R^n and its negation $\bar{R}^n$[3].

The set of terms is defined as in Definition 2.3.

Definition 3.2. The expression φ is an *atomic formula* iff one of the tree cases holds:

1. $\varphi \equiv P(t_1, ..., t_n)$ for some n-array predicate constant P and terms $t_1, ..., t_n$.
2. $\varphi \equiv t \in X$ for some term t and second order variable X.
3. $\varphi \equiv t \notin X$ for some term t and second order variable X.

As before $= (t_1, t_2)$ and $\neq (t_1, t_2)$ will be written as $t_1 = t_2$ and $t_1 \neq t_2$. Now we will define the language $\mathcal{L}_{\mathsf{T}}(\tau_{\text{nt}})$.

[3] For every set just one symbol

Remark 3.3. Note that the restriction to one term in case 2 and 3 in Definition 3.2 is not essential, because all primitive recursive functions are in the language n-tuples of terms can be coded by standard methods.

Definition 3.4. The set $\mathcal{L}_{\mathsf{T}}(\tau_{\mathrm{nt}})$ will be now defined in the following way:

1. If φ is an atomic formula, then $\varphi \in \mathcal{L}_{\mathsf{T}}(\tau_{\mathrm{nt}})$.
2. If $\varphi, \psi \in \mathcal{L}_{\mathsf{T}}(\tau_{\mathrm{nt}})$ then $(\varphi \wedge \psi)$ and $(\varphi \vee \psi)$ in $\mathcal{L}_{\mathsf{T}}(\tau_{\mathrm{nt}})$.
3. If $\varphi(a) \in \mathcal{L}_{\mathsf{T}}(\tau_{\mathrm{nt}})$ and the bound variable x does not occur in $\varphi(a)$ then $\exists x(\varphi(x)), \forall x(\varphi(x)) \in \mathcal{L}_{\mathsf{T}}(\tau_{\mathrm{nt}})$ where $\varphi(x)$ denotes $\varphi(a)[a/x]$.

We call the elements of $\mathcal{L}_{\mathsf{T}}(\tau_{\mathrm{nt}})$ *formulas* and formulas without free first order variables *(pseudo)* Π^1_1*-sentences*. Formulas without second order variables are called *arithmetical*.

Remark 3.5. Note that we imagine the free second order variables in the formulas as for all quantified. This is also the reasons why we call formulas without free first order variables (pseudo) Π^1_1-sentences.

Because negation is not a primitive symbol we have to define it:

Definition 3.6. For every formula $\varphi \in \mathcal{L}_{\mathsf{T}}(\tau_{\mathrm{nt}})$ we define $\neg\varphi$ inductively over the definition of $\mathcal{L}_{\mathsf{T}}(\tau_{nt})$.

1. If $\varphi \equiv t_1 = t_2$, then $\neg\varphi :\equiv t_1 \neq t_2$.
2. If $\varphi \equiv t_1 \neq t_2$, then $\neg\varphi :\equiv t_1 = t_2$.
3. If $\varphi \equiv t \in X$, then $\neg\varphi :\equiv t \notin X$.
4. If $\varphi \equiv t \notin X$, then $\neg\varphi :\equiv t \in X$.
5. If $\varphi \equiv R^n(t_1, ..., t_n)$ for a predicate constant correspond to the primitive recursive relation $R^{\mathbb{N}}$, then $\neg\varphi :\equiv \bar{R}^n(t_1, ..., t_n)$.
6. If $\varphi \equiv \bar{R}^n(t_1, ..., t_n)$ for a predicate constant correspond to the complement of the primitive recursive relation $R^{\mathbb{N}}$, then $\neg\varphi :\equiv R^n(t_1, ..., t_n)$.
7. If $\varphi \equiv \psi \wedge \sigma$, then $\neg\varphi :\equiv \neg\psi \vee \neg\sigma$.
8. If $\varphi \equiv \psi \vee \sigma$, then $\neg\varphi :\equiv \neg\psi \wedge \neg\sigma$.
9. If $\varphi \equiv \forall x\psi(x)$, then $\neg\varphi :\equiv \exists x\neg\psi(x)$.
10. If $\varphi \equiv \exists x\psi(x)$, then $\neg\varphi :\equiv \forall x\neg\psi(x)$.

Note that $\neg\neg\varphi \equiv \varphi$ holds literally. (I.e., not just as logical equivalence.) In the rest of the chapter finite sets of formulas will be denoted by capital greek letters like Γ, Δ, Λ. They are interpreted as finite disjunctions of the included formulas. We denote $\Gamma \cup \{\varphi\}$ by Γ, φ and omit the set braces if $\Gamma = \{\varphi\}$.

3.2 A Tait-Calculus for an Arithmetical Theory

We introduce formal systems using the finite Tait-language in this section. They are essential **PRA** where the induction schema is replaced by

$$\varphi(c_1) \wedge ... \wedge \varphi(c_m) \wedge \forall \vec{x}[\varphi(x_1) \wedge ... \wedge \varphi(x_n) \rightarrow \varphi(f(x_1, ..., x_n))] \rightarrow \forall x \varphi(x).$$

First let us define a set AX of finite sets of formulas, called axioms.

Definition 3.7. The set AX is the collection of following sets of formulas.

1. Logical Axioms: Every set of the form $\{\varphi, \neg\varphi\}$, where $\varphi \in \mathcal{L}_{\mathsf{T}}(\tau_{\mathrm{nt}})$ is atomic, is an axiom.
2. Mathematical Axioms: Each of the following is an axiom.[4]
   (a) $\forall x(0 \neq S(x))$
   (b) $\forall x \forall y(S(x) \neq S(y) \vee x = y)$
   (c) The defining universal formulas for all primitive recursive function constants and predicate constants.
   (d) $\bar{n} = S^n(0)$ for every $n \in \mathbb{N}$.
3. Identity property: For every n-array function symbol $f \in \tau_{\mathrm{nt}}$ and every $\varphi(\vec{a}) \in \mathcal{L}_{\mathsf{T}}(\tau_{\mathrm{nt}})$ finite sets of the following formulas are axioms.

$$\begin{aligned}
&\forall x(x = x)\\
&\forall x \forall y(x \neq y \vee y = x)\\
&\forall x \forall y \forall z(x \neq y \vee y \neq z \vee x = z)\\
&\forall \vec{x} \forall \vec{y}(x_1 \neq y_1 \vee ... \vee x_n \neq y_n \vee f(\vec{x}) = f(\vec{y}))\\
&\forall \vec{x} \forall \vec{y}(x_1 \neq y_1 \vee ... \vee x_n \neq y_n \vee \neg\varphi(\vec{a})[\vec{a}/\vec{x}] \vee \varphi(\vec{a})[\vec{a}/\vec{y}])
\end{aligned}$$

Remark 3.8. Because $\mathcal{L}_{\mathsf{T}}(\tau_{\mathrm{nt}})$ is primitive recursive, AX is as well.

I will use the notation $t^{\mathbb{N}}$ to denote the evaluation of a term $t \in \mathcal{L}_{\mathsf{T}}(\tau_{\mathrm{nt}})$ in the standard interpretation.

Definition 3.9. The Tait-Calculus TA_f
Assume $f \in \tau_{\mathrm{nt}}$ is an m-array primitive recursive function constant and $c_1, ..., c_l \in \tau_{\mathrm{nt}}$ are individual constants such that every $n \in \mathbb{N}$ is equal to a composition of $f^{\mathbb{N}}$ and the elements $c_1^{\mathbb{N}}, ..., c_l^{\mathbb{N}}$.

1. Axiom Rule:

$$\Gamma \cup \Delta$$

   where $\Delta \in AX$ and Γ arbitrary.

[4]Recall that we write φ instead of $\{\varphi\}$.

2. Conjunction Rule:
$$\frac{\Gamma, \varphi \quad \Gamma, \psi}{\Gamma, \varphi \wedge \psi}$$

3. Disjunction Rules:
$$\frac{\Gamma, \varphi}{\Gamma, \varphi \vee \psi}$$
$$\frac{\Gamma, \psi}{\Gamma, \varphi \vee \psi}$$

4. Generalisation Rule:
$$\frac{\Gamma, \varphi(a)}{\Gamma, \forall x \varphi(x)}$$
where a is a free variable which does not occur in the lower set of formulas, here a is called the *eigenvariable*.

5. Existence Rule:
$$\frac{\Gamma, \varphi(t)}{\Gamma, \exists x \varphi(x)}$$
where t is an arbitrary term.

6. Cut Rule:
$$\frac{\Gamma, \varphi \quad \Gamma, \neg\varphi}{\Gamma}$$
where φ is called the *cut formula*.

7. Induction:
$$\frac{\Gamma, \varphi(c_1) \quad ... \quad \Gamma, \varphi(c_l) \quad \Gamma, \neg\varphi(a_1) \wedge ... \wedge \neg\varphi(a_m) \vee \varphi(f(a_1, ..., a_m))}{\Gamma, \varphi(t)}$$

We call the formulas in Γ the *side formulas*, the formulas shown in the premisses the *auxiliary formulas* and the formulas shown in the conclusion *principal formulas*.

Remark 3.10. Obviously TA_S (where S is the successor function) can be seen as **PA** extended by axioms for all primitive recursive functions and free second order variables. This extension is a conservative one (see [8]).

Definition 3.11. A *deduction* D of Λ in TA_f, $D \vdash \Lambda$, is a finite, rooted, ordered tree[5], whose nodes are labelled by a finite set of formulas and (a tag[6] of) a deduction rule, satisfying the following conditions:

[5]If two notes are comparable, then we call the note closer to the root “below” or “successor” of the other.

[6]By a tag we mean a symbol for each set of inferences (rule), e.g. the string "cut" for the cut rule.

1. All leaves are labelled by an instance of the axiom rule and the tag of the axiom rule.

2. If a node, which is not a leaf, is labelled by a set $\Gamma \cup \Delta$ and a deduction rule R of TA_f and if its immediate predecessors are labelled by the sets Γ, φ_i for $i \in I$, then the inference

$$\frac{\Gamma, \varphi_i}{\Gamma \cup \Delta} \text{ (for all i} \in \text{I)}$$

is an instance of the deduction rule R.

3. The finite set of formulas of the root is Λ.

We say TA_f *deduces* Λ, in symbols $\mathsf{TA}_f \vdash \Lambda$ iff there is a deduction D in TA_f such that $D \vdash \Lambda$.

Later, an infinite Tait-calculus (where index sets are possible infinite) will be given into which TA_f can be embedded. This infinite Tait-calculus has full-cutelimination.

3.3 An Infinite Language and Tait-Calculus

In this section we define an infinite version of the Tait-language. Not surprisingly one can replace quantification and first order variables whit infinite conjunctions and disjunctions.

Definition 3.12. The set of *primitive symbols* of $\mathcal{L}_\infty(\tau_{\text{nt}})$ consists of the following:

1. Logical Symbols
   (a) Logical connectives: $\bigwedge, \bigvee$
   (b) Second order variables: $X_0, X_1, X_2, ...$
   (c) Relationship symbols: $\in, \notin$
   (d) Brackets: $(,)$

2. Constants: The set τ_{nt} of constants consists of
   (a) An individual constant for every natural number: $\bar{0}, \bar{1}, \bar{2}, \bar{3}, ...$
   (b) The 1-array function constant S "successor" and for every other n-array primitive recursive function with $n \in \omega$ a n-array function constant: $f_0^n, f_1^n, f_2^n, ...$
   (c) Two 2-array Predicate constant $=$ and $\neq$. And for every primitive recursive n-array relation exactly one predicate constant R^n and its negation $\bar{R}^n$.

The set of terms is defined as in Definition 2.3 apart from the fact that the language does not have first order variables. Atomic formulas are defined as before in Definition 3.2.

Definition 3.13. The set $\mathcal{L}_\infty(\tau_{\mathrm{nt}})$ is defined by the following induction:

1. If φ is a atomic formula then $\varphi \in \mathcal{L}_\infty(\tau_{\mathrm{nt}})$.
2. If I is a finite set and $\varphi_i \in \mathcal{L}_\infty(\tau_{\mathrm{nt}})$ for all $i \in I$, then
$$\bigvee_{i\in I} \varphi_i \text{ and } \bigwedge_{i\in I} \varphi_i$$
are in $\mathcal{L}_\infty(\tau_{\mathrm{nt}})$.
3. If $I = \mathbb{N}$ and $\varphi_i \equiv \psi(\bar{i}) \in \mathcal{L}_\infty(\tau_{\mathrm{nt}})$ [7] for all $i \in I$, then
$$\bigvee_{i\in I} \varphi_i \text{ and } \bigwedge_{i\in I} \varphi_i$$
are in $\mathcal{L}_\infty(\tau_{\mathrm{nt}})$.

Remark 3.14. Note that $\mathcal{L}_\infty(\tau_{\mathrm{nt}})$ is a "constructively given complete species" in the sense of [15].

As in the case of $\mathcal{L}_{\mathsf{T}}(\tau_{\mathrm{nt}})$ we have to introduce negation for $\mathcal{L}_\infty(\tau_{\mathrm{nt}})$.

Definition 3.15. For every formula $\varphi \in \mathcal{L}_\infty(\tau_{nt})$ we define $\neg\varphi$ inductively over the definition of $\mathcal{L}_\infty(\tau_{nt})$.

1. If $\varphi \equiv t_1 = t_2$, then $\neg\varphi :\equiv t_1 \neq t_2$.
2. If $\varphi \equiv t_1 \neq t_2$, then $\neg\varphi :\equiv t_1 = t_2$.
3. If $\varphi \equiv t \in X$, then $\neg\varphi :\equiv t \notin X$.
4. If $\varphi \equiv t \notin X$, then $\neg\varphi :\equiv t \in X$.
5. If $\varphi \equiv R^n(t_1, ..., t_n)$, then $\neg\varphi :\equiv \bar{R}^n(t_1, ..., t_n)$.
6. If $\varphi \equiv \bar{R}^n(t_1, ..., t_n)$, then $\neg\varphi :\equiv R^n(t_1, ..., t_n)$.
7. If $\varphi \equiv \bigwedge_{i\in I} \varphi_i$, then $\neg\varphi :\equiv \bigvee_{i\in I} \neg\varphi_i$.
8. If $\varphi \equiv \bigvee_{i\in I} \varphi_i$, then $\neg\varphi :\equiv \bigwedge_{i\in I} \neg\varphi_i$.

As before $\neg\neg\varphi \equiv \varphi$ holds. With this language a infinite Tait-calculus can be given.

Definition 3.16. The set of finite set of formulas AX_∞ is defined as follows:

[7] Here $\psi(\bar{i})$ means that the constant $\bar{i}$ occurs in φ_i and that for every $i, j \in I$ $\varphi_i[\bar{i}/\bar{j}] \equiv \varphi_j$

1. If $\varphi \in \mathcal{L}_\infty(\tau_{\mathrm{nt}})$ is atomic arithmetical and $\mathbb{N}$-true, then $\{\varphi\} \in AX_\infty$.
2. If $t^{\mathbb{N}} = s^{\mathbb{N}}$, then $\{t \notin X, s \in X\} \in AX_\infty$.

Remark 3.17. Some times sets as AX_∞ are called axiom systems. The most general notion of an axiom system for a Tait-calculus is a set $\mathbf{AX}$ of finite non empty sets of atomic formulas which satisfies the *intersection property*[15]:
If $\Delta, \varphi \in \mathbf{AX}$ and $\Delta', \neg\varphi \in \mathbf{AX}$ then there is an $\Lambda \subset \Delta \cup \Delta'$ such that $\Lambda \in \mathbf{AX}$.
It should be obvious that AX_∞ has this property. So we can apply Tait's proofs in [15] to NT_∞. It is also a logical complete axiom systems which means that for every atomic formula φ there is a non empty subset of $\{\varphi, \neg\varphi\}$ in $\mathbf{AX}$ [15].

Definition 3.18. The Tait-Calculus NT_∞

1. Axiom Rules:
$$\Gamma \cup \Delta$$
for $\Delta \in AX_\infty$ and Γ arbitrary.
2. Disjunction:
$$\frac{\Gamma, \varphi_i}{\Gamma, \bigvee_{i \in I} \varphi_i} \text{ (for some } i \in I)$$
3. Conjunction:[8]
$$\frac{\Gamma, \varphi_i}{\Gamma, \bigwedge_{i \in I} \varphi_i} \text{ (for all } i \in I)$$
4. Cut:
$$\frac{\Gamma, \varphi \quad \Gamma, \neg\varphi}{\Gamma}$$
where φ is called the *cut formula*.

We call the formulas in Γ the *side formulas*, the formulas shown in the premisses the *auxiliary formulas* and the formulas shown in the conclusion *principal formulas*.

Remark 3.19.
1. Note that NT_∞ is an infinite propositional calculus. The deductions of NT_∞ (see Definition 3.20) are not recursive enumerable even the axioms are primitive recursive.
2. From the definition of the axiom rule it should be clear where are the difference to Gentzen's method. Here we need all standard-model-true atomic sentences. So from the point of view of Definition 1.48 this can leads into troubles if one uses a language where the predicate and function constants do not have a primitive recursive interpretation.

[8]Note that the instances of this deduction rule can be inferences with infinitely many premisses if the principal formula of the inference has the form $\bigwedge_{i \in \mathbb{N}} \varphi_i$.

A tree is called well-founded iff it has no infinite paths.

Definition 3.20. A *deduction* D of Λ in NT_∞, $D \vdash \Lambda$, is a (possible infinite) well-founded, rooted, ordered tree[9], whose nodes are labelled by a finite set of formulas and (a tag of) a deduction rule, satisfying the following conditions:

1. All leaves are labelled by an instance of one of the axiom rules and the tag of this axiom rule.
2. If a node, which is not a leaf, is labelled by a set $\Gamma \cup \Delta$ and a deduction rule R of NT_∞ and if its immediate predecessors are labelled by the sets Γ, φ_j for $j \in J$, then the inference[10]
$$\frac{\Gamma, \varphi_j}{\Gamma \cup \Delta} \text{ (for all j} \in \text{J)}$$
is an instance of the deduction rule R.
3. The finite set of formulas of the root is Λ.

We say NT_∞ *deduces* Λ, in symbols $\mathsf{NT}_\infty \vdash \Lambda$ iff there is a deduction D in NT_∞ such that $D \vdash \Lambda$.

The next definition will extend the concept of rank and will give a way to count the application of deduction rules. The rank for finite formulas is defined like in Chapter 2, also ordinals will be denoted as before.

Definition 3.21. Assume $\varphi \in \mathcal{L}_\infty(\tau_{\mathrm{nt}})$. The rank of a formula, $\mathrm{rank}(\varphi) \leq \alpha$, is inductively defined as follows:

1. If φ is atomic, then $\mathrm{rank}(\varphi) \leq \alpha$ for all $\alpha < \omega_1$.
2. If $\varphi \equiv \bigwedge_{i \in I} \varphi_i$ with $\mathrm{rank}(\varphi_i) \leq \alpha_i$ and $\alpha_i < \alpha$ for all $i \in I$, then $\mathrm{rank}(\varphi) \leq \alpha$.
3. If $\varphi \equiv \bigvee_{i \in I} \varphi_i$ with $\mathrm{rank}(\varphi_i) \leq \alpha_i$ and $\alpha_i < \alpha$ for all $i \in I$, then $\mathrm{rank}(\varphi) \leq \alpha$.

Remark 3.22. From the definition it should be obvious that $\mathrm{rank}(\varphi) = \mathrm{rank}(\neg\varphi)$.

The next definition gives the rank of a derivation in the two Tait-calculi mentioned before. From now on we write T if we speak about TA_f and NT_∞ simultaneously.

---

[9]If two notes are comparable, then we call the note closer to the root "below" or "successor" of the other.

[10]Do not be confused. In general, the set J does not coincide with the index sets of the principal formulas in applications of the rules Conjunction and Disjunction. J is only an index set which enumerates the labels of the predecessors (premisses) in the tree.

Definition 3.23. Assume D is a derivation in T. Then the rank$(D) \leq \alpha$ is defined as follows:

1. If D is an instance of an axiom rule, then rank$(D) \leq \alpha$ for all $\alpha < \omega_1$.
2. If the last deduction rule of D is a rule with the following form

$$\frac{\Gamma, \varphi_j}{\Gamma \cup \Delta} \text{ (for all j} \in \text{J)}$$

   and D_j are the subdeductions of D with the form

$$\begin{array}{c} \vdots \\ \Gamma, \varphi_j \end{array}$$

   such that rank$(D_j) \leq \alpha_j$ with $\alpha_j < \alpha$ for all $j \in J$, then rank$(D) \leq \alpha$.

Obviously in the case of TA_f the rank of a deductions is always finite. The following definition gives a weighted way to count the cuts in a deduction.

Definition 3.24. Assume D is a deduction in T. The *cut degree* of D, or cd(D), is $\leq \alpha$ iff all cut formulas φ in D have rank$(\varphi) < \alpha$.

So by definition cd$(D) \leq 0$ means there is no application of the cut rule in D.

Notation 3.25. $D \vdash \Delta[\alpha, \beta]$ iff $D \vdash \Delta$ and rank$(D) \leq \alpha$ and cd$(D) \leq \beta$.

From the previous definition it should be clear what $\mathsf{T} \vdash \Delta[\alpha, \beta]$ means. The notation $D \cup \Delta$ will denote the deduction in T which is obtained from D by extending every set of formulas in D by the set Δ. The next two lemmas should be obvious from Definition 3.18.

Lemma 3.26. *If* $D \vdash \Gamma[\alpha, \beta]$, *then* $D \cup \Delta \vdash \Gamma \cup \Delta[\alpha, \beta]$.

Lemma 3.27. *If* $\mathsf{NT}_\infty \vdash \Gamma, \varphi(t)[\alpha, \beta]$ *and* $t^{\mathbb{N}} = s^{\mathbb{N}}$, *then* $\mathsf{NT}_\infty \vdash \Gamma, \varphi(s)[\alpha, \beta]$.

We call a deduction rule permissible iff it does not extend the set of derivable syntactical objects. The next definition gives a concept which lead to some permissible deduction rules for NT_∞

Definition 3.28. A *reduction* of a $\varphi \in \mathcal{L}_\infty(\tau_{\text{nt}})$ is a finite set Θ of formulas such that:

1. If φ is an atom which is not an axiom, then Θ is arbitrary.
2. If φ is an atom which is an axiom, then φ has no reduction.
3. If $\varphi \equiv \bigwedge_{i \in I} \varphi_i$, then Θ is an reduction of φ iff there is an $i \in I$ such that $\varphi_i \in \Theta$, .

4. If $\varphi \equiv \bigvee_{i \in I} \varphi_i$ such that I is finite, then every $\Theta \supseteq \{\varphi_i | i \in I\}$ is an reduction of φ.

5. If $\varphi \equiv \bigvee_{i \in I} \varphi_i$ such that I is not finite, then φ has no reduction.

With this we can formulate the next lemma.

Lemma 3.29. *Assume Θ is an reduction of φ.*
If $\mathsf{NT}_\infty \vdash \Gamma, \varphi[\alpha, \beta]$, then $\mathsf{NT}_\infty \vdash \Gamma \cup \Theta[\alpha, \beta]$

Proof. See [15, p.212]. □

The next lemma gives a bound of the rank of a cut-free deduction of an instance of the tertium non datur.

Lemma 3.30. *If* $\mathrm{rank}(\varphi) \preceq \alpha$, *then* $\mathsf{NT}_\infty \vdash \{\varphi, \neg\varphi\}[2 \cdot \alpha, 0]$.

Proof. See [15, p.212]. □

Now we formulate the main lemma, from which cut-elimination can be easily proven.

Lemma 3.31. ***Elimination Lemma***
Assume $\mathsf{NT}_\infty \vdash \Gamma, \varphi[\alpha, \gamma]$ and $\mathsf{NT}_\infty \vdash \Delta, \neg\varphi[\beta, \gamma]$, where $\mathrm{rank}(\varphi) \preceq \gamma$, *then* $\mathsf{NT}_\infty \vdash \Gamma \cup \Delta[\alpha \sharp \beta, \gamma]$

Proof. See [15, p.213]. □

This gives us cut-elimination together with an upper bound of the rank-rising after cut-elimination in terms of the χ_T-function (see Section 1.1.4).

Theorem 3.32. ***Elimination Theorem***
If $\mathsf{NT}_\infty \vdash \Delta[\alpha, \beta + \omega^\gamma]$, then $\mathsf{NT}_\infty \vdash \Delta[\chi_\mathsf{T}(\gamma, \alpha), \beta]$.

Proof. See [15, p.214]. □

Remark 3.33. If one read the proof in [15] carefully one sees that the operations can be make primitive recursive.

Before the next corollary can be formulated the following notation has to be introduced.

$$2_0(\alpha) := \alpha$$

$$2_{n+1}(\alpha) := 2^{2_n(\alpha)}.$$

Corollary 3.34. *If $\mathsf{NT}_\infty \vdash \Delta[\alpha, \beta + n]$, then $\mathsf{NT}_\infty \vdash \Delta[2_n(\alpha), \beta]$.*

Proof. Imagine $n = \omega^0 + ... + \omega^0$ and apply Theorem 3.32 n-times. □

We get the next corollary just by observing that in a cut-free deduction it is not possible to lose any formulas.

Corollary 3.35. *Every in NT_∞ derivable set of atomic formulas include an axiom.*

Remark 3.36. The syntactical objects, a Tait-calculus is dealing with, are finite sets of formulas. So by the discussion above Definition 2.46 a consistency proof have to show that there are finite sets of formulas which are not derivable. This observation makes cut-elimination into a consistency proof, because of Corollary 3.35 which leads to:

$$\text{If } \varphi \text{ is atomic and } \mathbb{N}\text{-false, then } \mathsf{NT}_\infty \nvdash \{\varphi\}.$$

3.4 An Embedding and an application of Cut-Elimination

In general the calculi TA_f do not have cut-elimination. For example, this is the case for TA_S.

Assume toward a contradiction that TA_S does have cut-elimination, then also the subsystem of TA_S (without second order variables) which include only the defining formulas for successor, addition and multiplication has cut-elimination. This restricted system is essential **PA**. So we obtain that **PA** has cut-elimination. It is a well known fact that in **PA** the exponential function is definable (see [2, Sec.3.8]), say via φ, and that $\mathbf{PA} \vdash \exists x\varphi(\bar{n}, x)$ for every $n \in \mathbb{N}$. Since **PA** has cut-elimination there is an Herbrand-Disjunction $\bigvee_{i=0}^{m} \varphi(\bar{n}, t_i)$ for every $n \in \mathbb{N}$. Because the language just contains successor, addition and multiplication as function symbols for all $0 \leq i \leq m$ the term t_i is a polynomial. This leads to the contradiction that exponentiation can be approximate by polynomials.

So instead of using cut-elimination in TA_f, we embed TA_f into NT_∞ and use cut-elimination there. To do this, we first give a embedding of $\mathcal{L}_{\mathsf{T}}(\tau_{\mathrm{nt}})$ into $\mathcal{L}_\infty(\tau_{\mathrm{nt}})$ by expressing quantification by infinite formulas.

A *syntactical assignment* is a function $\iota : \{a_1, a_2, a_3, ...\} \to \{\bar{0}, \bar{1}, \bar{2}, ...\}$. The next definition will show how a syntactical assignment can be extended to an embedding of $\mathcal{L}_{\mathsf{T}}(\tau_{\mathrm{nt}})$ into $\mathcal{L}_\infty(\tau_{\mathrm{nt}})$.

Definition 3.37. Assume ι is an syntactical assignment. An *embedding* $\iota : \mathcal{L}_{\mathsf{T}}(\tau_{\mathrm{nt}}) \to \mathcal{L}_\infty(\tau_{\mathrm{nt}})$ is defined as follows.

1. If $\varphi \equiv \psi(a_1, ..., a_m)$ is atomic and all free variables are shown, then $\iota(\varphi) \equiv \psi(\iota(a_1), ..., \iota(a_m))$.
2. If $\varphi \equiv \psi \wedge \sigma$, then $\iota(\varphi) \equiv \bigwedge_{i\in I} \varphi_i$ where $I = \{0, 1\}$, $\varphi_0 \equiv \iota(\psi)$ and $\varphi_1 \equiv \iota(\sigma)$.
3. If $\varphi \equiv \psi \vee \sigma$, then $\iota(\varphi) \equiv \bigvee_{i\in I} \varphi_i$ where $I = \{0, 1\}$, $\varphi_0 \equiv \iota(\psi)$ and $\varphi_1 \equiv \iota(\sigma)$.

4. If $\varphi \equiv \forall x(\psi(x))$, then $\iota(\varphi) \equiv \bigwedge_{i \in I} \varphi_i$ where $I = \mathbb{N}$, $\varphi_i \equiv \iota(\psi_i)$ and $\psi_i \equiv \psi(x)[x/\bar{i}]$.
5. If $\varphi \equiv \exists x(\psi(x))$, then $\iota(\varphi) \equiv \bigvee_{i \in I} \varphi_i$ where $I = \mathbb{N}$, $\varphi_i \equiv \iota(\psi_i)$ and $\psi_i \equiv \psi(x)[x/\bar{i}]$.

For a finite set of formulas Δ in $\mathcal{L}_{\mathsf{T}}(\tau_{\mathrm{nt}})$ the embedding is denoted by $\iota(\Delta)$.

Obviously $\mathrm{rank}(\iota(\varphi)) \prec \omega$ for all $\varphi \in \mathcal{L}_{\mathsf{T}}(\tau_{\mathrm{nt}})$. Also clear from definition of embedding is the next lemma.

Lemma 3.38. *Assume $\varphi \equiv \psi(a_1, ..., a_n) \in \mathcal{L}_{\mathsf{T}}(\tau_{nt})$ where all free variables are shown, ι_1 and ι_2 are embeddings. Then holds:*

$$\text{If } \iota_1(a_i) = \iota_2(a_i) \text{ for all } 1 \leq i \leq n, \text{ then } \iota_1(\varphi) = \iota_2(\varphi)$$

Now we start to show some the results which together will prove the Embedding Theorem (see Theorem 3.41).

Lemma 3.39. *If $\Delta \in AX$, then $\mathsf{NT}_\infty \vdash \iota(\Delta)[\omega, 0]$ for all embeddings ι.*

Proof. The proof is a case distinction by the Definition 3.7 of AX.

1. Δ is a logical axiom, then the result follows from Lemma 3.30.
2. Δ is a mathematical axiom, then $\Delta = \{\varphi\}$ for an universal sentence $\varphi \equiv \forall \vec{x} \psi(\vec{x})$. Since φ is $\mathbb{N}$-true, $\psi(\vec{\bar{n}})$ for every $\vec{n} \in \mathbb{N}^{<\omega}$ is $\mathbb{N}$-true and therefore an axiom of NT_∞ so we get $\iota(\Delta)$ by one applications of Conjunction. The result follows by Lemma 3.38.
3. Δ is an identity axiom. For all instances of the first four identity axiom the proof is like in Case 2, for the last one observe that

   $$\mathsf{NT}_\infty \vdash \{\bar{n}_1 \neq \bar{m}_1, ..., \bar{n}_k \neq \bar{m}_k, \iota(\neg\varphi(\vec{\bar{n}})), \iota(\varphi(\vec{\bar{m}}))\}[l, 0]$$

   for $l \in \omega$ holds if $n_i \neq m_i$ for one $1 \leq i \leq k$ by the axiom rule of NT_∞ and Lemma 3.26 and if for all $1 \leq i \leq k$ $n_i = m_i$ then by Lemma 3.30. The result follows by finite applications of Conjunction and Lemma 3.38. □

Lemma 3.40. *If $\mathsf{TA}_f \vdash \Delta$ without an application of the induction rule, then $\mathsf{NT}_\infty \vdash \iota(\Delta)[\omega + n, m]$ for some $n, m \in \omega$ and all embeddings ι.*

Proof. From $\mathsf{TA}_f \vdash \Delta$ it follows that there is a deduction D in TA_f such that $D \vdash \Delta$. Since D is in TA_f by Definition 3.23 follows $\mathrm{rank}(D) < \omega$ and $\mathrm{cd}(D) < \omega$. So there are $n, m \in \omega$ such that $D \vdash \Delta[n, m]$. This deduction D can be used to construct a deduction in NT_∞ by induction on the applications of deduction rules.
Induction Start: By Lemma 3.39 we get $D_{0,\infty} \vdash \iota(\Lambda)[\omega, 0]$ for all Λ in D which are an application of the axiom rule of TA_f and all ι.
Induction Step: There are just two cases are given the rest is analogues, note that the case of a cut is the most trivial one.

1. Assume in D is an application of the conjunction rule, say

$$\frac{\Gamma, \varphi \quad \Gamma, \psi}{\Gamma, \varphi \wedge \psi}$$

and there are already deductions $D_{l_0,\infty} \vdash \Gamma, \iota(\varphi)[\omega+k_0, q_0]$ and $D_{l_1,\infty} \vdash \Gamma, \iota(\psi)[\omega + k_1, q_1]$ in NT_∞ for all ι. Then we define $I := \{0, 1\}$ and with an application of conjunction from NT_∞ we get

$$D_{l,\infty} \vdash \Gamma, \bigwedge\{\iota(\varphi), \iota(\psi)\}[\omega + k, q]$$

such that $k = \max\{k_0, k_1\} + 1$ and $q = \max\{q_0, q_1\}$ for all ι.

2. Assume in D is an application of the generalisation rule, say

$$\frac{\Gamma, \varphi(a)}{\Gamma, \forall x\varphi(x).}$$

Because of Lemma 3.38 and the fact that a free variable is involved we get from the induction hypothesis for all $i \in \omega$ deductions $D_{l_i,\infty} \vdash \Gamma, \iota_i(\varphi(a))[\omega + k_i, q_i]$, where $\iota_i(a) = \bar{i}$ and all k_i and q_i are the same. So by an application of conjunction from NT_∞ with $I = \mathbb{N}$ we get

$$D_{l,\infty} \vdash \Gamma, \bigwedge\{\iota(\varphi(\bar{i}))|i \in \omega\}[\omega + k, q]$$

where $k = k_0 + 1$ and $q = q_0$.

From this we get a deduction $D_{n,\infty} \vdash \iota(\Delta)[\omega+n, m]$ where $n = \operatorname{rank}(D)$ □

Remember the notation of an f-term in Notation 2.17.

Theorem 3.41. ***Embedding Theorem***
If $\mathsf{TA}_f \vdash \Delta$, then $\mathsf{NT}_\infty \vdash \iota(\Delta)[\omega^2, q]$ for some $q \in \omega$ and all embeddings ι.

Proof. Like in the proof of Lemma 3.40 there is a D in TA_f such that $D \vdash \Delta[n, q]$ for $n, q \in \omega$. The result follows by induction from the case where the last deduction rule is an application of induction. So assume the last deduction rule in D have the form

$$\frac{\Gamma, \varphi(c_1) \quad ... \quad \Gamma, \varphi(c_l) \quad \Gamma, \neg\varphi(a_1) \vee ... \vee \neg\varphi(a_m) \vee \varphi(f(a_1, ..., a_m))}{\Gamma, \varphi(t)}.$$

By Lemma 3.40 and 3.29 we get $l+1$ deductions in NT_∞ with $\operatorname{rank}(D_{\infty,i}) \leq \omega + n_i$ and $\operatorname{cd}(D_{\infty,i}) \leq q_i$ such that:

$$\begin{array}{cccc} D_{\infty,1} & & D_{\infty,l} & D_{\infty,l+1} \\ \vdots & & \vdots & \vdots \\ \iota(\Gamma), \iota(\varphi(c_1)); & ... ; & \iota(\Gamma), \iota(\varphi(c_l)); & \iota(\Gamma), \iota(\neg\varphi(a_1)), ..., \iota(\neg\varphi(a_m)), \iota(\varphi(f(a_1, ..., a_m))). \end{array}$$

Because there are just constants for primitive recursive functions in τ_{nt} the problem $t^{\mathbb{N}} = k \in \mathbb{N}$ is an primitive recursive one. Because of Definition 3.9 there is an f-term $\bar{t}$ such that $\bar{t}^{\mathbb{N}} = k$. By induction over the complexity of $\bar{t}$ it is possible to show that $\mathsf{NT}_\infty \vdash \iota(\Gamma), \iota(\varphi(\bar{t}))[p \cdot \omega, q]$ for $m, p < \omega$.
Induction start: Assume $\bar{t} \equiv c_i$ for $1 \leq i \leq l$. The claim follows by deduction $D_{\infty,i}$.
Induction step: Assume $\bar{t} \equiv f(t_1, ..., t_m)$ and we already have

$$\mathsf{NT}_\infty \vdash \iota(\Gamma), \iota(\varphi(t_i))[p_i \cdot \omega, q_i]$$

for $1 \leq i \leq m$.
From $D_{\infty,l+1}$ and Lemma 3.27 we obtain the deduction

$$\begin{array}{c} D_{\infty,l+1} \\ \vdots \\ \iota(\Gamma), \iota(\neg\varphi(t_1)), ..., \iota(\neg\varphi(t_m)), \iota(\varphi(f(t_1, ..., t_m))) \end{array}$$

which is possible because in all t_i no free variable occurs. After m applications of the cut rule we get

$$\mathsf{NT}_\infty \vdash \iota(\Gamma), \iota(\varphi(\bar{t}))[p \cdot \omega, q]$$

where $p = \max\{p_1, ..., p_m\} + 1 < \omega$ and $q = \max\{q_1, ..., q_m, \mathrm{rank}(\iota(\varphi(\bar{t})))\} < \omega$.
Since $\bar{t}^{\mathbb{N}} = k = t^{\mathbb{N}}$ this leads with Lemma 3.27 to $\mathsf{NT}_\infty \vdash \iota(\Gamma), \iota(\varphi(t))[p \cdot \omega, q]$.
So an easy induction on the applications of the induction rule in a deduction D of TA_f proves the result. □

Now we can apply cut elimination on the embedded deductions.
With Corollary 3.34 together with the fact that $2_n(\omega^2) < \varepsilon_0 = \chi_{\mathsf{T}}(1, 0)$ for all $n \in \omega$ we get:

Corollary 3.42. *If* $\mathsf{TA}_f \vdash \Delta$*, then* $\mathsf{NT}_\infty \vdash \iota(\Delta)[\alpha, 0]$ *for some* $\alpha < \varepsilon_0$ *and all embeddings* ι*.*

3.5 The Π^1_1-Ordinal of a Tait-calculus

An application of the embedding-method is to calculate the Π^1_1-Ordinal of the TA_f which is defined in this section.

We define the truth of a formula $\varphi(\vec{X})$ which contains second order variables as follows:

$$\mathbb{N} \models \varphi(\vec{X}) :\Leftrightarrow \mathbb{N} \models \forall \vec{X} \varphi(\vec{X}).$$

Here we mean the standard definition of truth in a second order model. As usual, a finite set of formulas Δ is interpreted as a disjunction of its elements: so we set

$$\mathbb{N} \models \Delta \text{ iff } \mathbb{N} \models \bigvee_{\varphi \in \Delta} \varphi.$$

Definition 3.43. Assume $\mathbb{N}$ is the standard model of arithmetic, $\alpha \in \mathbf{ON}$[11] and φ is a Π^1_1-sentence of $\mathcal{L}_{\mathsf{T}}(\tau_{\mathrm{nt}})$. Then the $\mathbb{N} \models^\alpha \Delta$ is defined as follows:

1. If φ is true, atomic and arithmetical[12], then $\mathbb{N} \models^\alpha \Gamma, \varphi$ for all $\alpha < \omega_1$.
2. If $t^{\mathbb{N}} = s^{\mathbb{N}}$, then $\mathbb{N} \models^\alpha \Gamma, t \notin X, s \in X$ for all $\alpha < \omega_1$.
3. $\mathbb{N} \models^\alpha \Gamma, \psi_1 \wedge \psi_2$ iff $\mathbb{N} \models^{\alpha_i} \Gamma, \psi_1 \wedge \psi_2, \psi_i$ for $i \in \{1,2\}$ and $\alpha_1, \alpha_2 < \alpha$.
4. $\mathbb{N} \models^\alpha \Gamma, \psi_1 \vee \psi_2$ iff there is an $i \in \{1,2\}$ such that $\mathbb{N} \models^{\alpha_i} \Gamma, \psi_1 \vee \psi_2, \psi_i$ and $\alpha_i < \alpha$.
5. $\mathbb{N} \models^\alpha \Gamma, \forall x \psi(x)$ iff $\mathbb{N} \models^{\alpha_n} \Gamma, \forall x \psi(x), \psi(\bar{n})$ for all $n \in \mathbb{N}$ and $\alpha_n < \alpha$ for all $n \in \mathbb{N}$.
6. $\mathbb{N} \models^\alpha \Gamma, \exists x \psi(x)$ iff there is an $n \in \mathbb{N}$ such that $\mathbb{N} \models^\beta \Gamma, \exists x \psi(x), \psi(\bar{n})$ and $\beta < \alpha$.

The next definition gives the famous Church-Kleene-ordinal.

Definition 3.44. $\omega_1^{CK} := \bigcup\{\mathrm{otyp}(\prec) |\ \prec$ is a primitive recursive ordering on $\omega\}$

The next Theorem gives an important property of $\models^\alpha$.

Theorem 3.45. ***ω-Completeness Theorem***
For all Π^1_1-sentences $\varphi(\vec{X})$ we have:

$$\mathbb{N} \models \varphi(\vec{X}) \text{ iff there is an } \alpha < \omega_1^{CK} \text{ such that } \mathbb{N} \models^\alpha \varphi(\vec{X})$$

Proof. See [8, Sec.5.4]. □

Remark 3.46. The proof of Theorem 3.45 uses the fact that all functions in the language are primitive recursive.

Notation 3.47. For a formula φ let $\varphi^\triangledown$ denote the finite set that is obtained from φ by braking up all disjunction, i.e.,

$$\varphi^\triangledown := \begin{cases} \psi_1^\triangledown \cup \psi_2^\triangledown & \text{if} \quad \varphi \equiv \psi_1 \vee \psi_2 \\ \{\varphi\} & \qquad\text{otherwise.} \end{cases}$$

This definition leads to the obvious fact that

$$\text{if } \mathbb{N} \models^\alpha \varphi \text{ then } \mathbb{N} \models^\alpha \varphi^\triangledown.$$

Definition 3.48. Assume φ is a Π^1_1-sentence, then we define the *truth complexity* as

$$\mathrm{tc}(\varphi) := \min(\{\alpha | \mathbb{N} \models^\alpha \varphi^\triangledown\} \cup \{\omega_1^{CK}\}).$$

[11] Here we mean an ordinal, not an element of an ordinal notation system.
[12] No occurrence of second order variables.

By this definition we get the next corollary from Theorem 3.45.

Corollary 3.49. *For every Π^1_1-sentence φ holds:*

$$\mathbb{N} \models \varphi \text{ iff } \mathrm{tc}(\varphi) < \omega_1^{CK}$$

The next lemma is obvious from Definition 3.43 by an induction on the rank of an deduction in NT_∞.

Lemma 3.50. *Assume Δ is a set of Π^1_1-sentences (in $\mathcal{L}_\mathsf{T}(\tau_{nt})$) and ι as in Definition 3.37.*
If $\mathsf{NT}_\infty \vdash \iota(\Delta)[\alpha, 0]$, then there is an $\beta \leq \alpha$ such that $\mathbb{N} \models^\beta \Delta$.

Assume T is a finite Tait-calculus in a language $\mathcal{L}_\mathsf{T}(\tau_{\mathrm{nt}})$. If $\prec$ is a order relation on $A \subset \mathbb{N}$ definable in $\mathcal{L}_\mathsf{T}(\tau_{\mathrm{nt}})$ (such as primitive recursive order relations) we can formulate the sentence of transfinite induction with the help of second order variables as

$$TI(\prec): \quad \forall x[(\forall y \prec x)[y \in X] \to x \in X] \to \forall x[A(x) \to x \in X].$$

In the sentence above $(\forall y \prec x)$ and $A(x)$ are appropriations for there definitions in $\mathcal{L}_\mathsf{T}(\tau_{\mathrm{nt}})$. In the next definition a correct calculus is a calculus which deduct only true sentences (as TA_f).

Definition 3.51. Assume T is a correct finite Tait-calculus in a language $\mathcal{L}_\mathsf{T}(\tau_{\mathrm{nt}})$. Then is the otyp-*ordinal* and the Π^1_1-*ordinal of the calculus* T defined as follows:

$$\begin{aligned}
\|\mathsf{T}\|_{\mathrm{otyp}} &:= \sup\{\mathrm{otyp}(\prec) | (A, \prec) \text{ is primitive recursive and } \mathsf{T} \vdash TI(\prec)\} \\
\|\mathsf{T}\|_{\Pi^1_1} &:= \sup\{\mathrm{tc}(\varphi) | \varphi \text{ is a } \Pi^1_1\text{-sentence and} \mathsf{T} \vdash \varphi\}
\end{aligned}$$

Note that $\|\mathsf{T}\|_{\mathrm{otyp}} \leq \|\mathsf{T}\|_{\Pi^1_1}$ as [8] proves. From Corollary 3.49 we obtain

$$\|\mathsf{TA}_f\|_{\Pi^1_1} < \omega_1^{CK}.$$

With the use of the embedding-method presented in the previous sections of this chapter we can use Lemma 3.50 to show $\|\mathsf{TA}_f\|_{\Pi^1_1} \leq \varepsilon_0$. It seems to me that the lower bound of $\|\mathsf{T}\|_{\mathrm{otyp}}$ can be found by using the ordinal notation system $(\mathbf{ON}_{\varepsilon_0}, \prec_{\varepsilon_0})$ (as [8] do for $\mathbf{PA}$). By coding $(\mathbf{ON}_{\varepsilon_0}, \prec_{\varepsilon_0})$ into TA_f one can show that for all orderings $(\mathbf{ON}_\alpha, \prec_\alpha)$ with $\alpha \prec \varepsilon_0$ the calculus $\mathsf{TA}_f \vdash TI(\prec_\alpha)$. This would lead to

$$\varepsilon_0 \leq \|\mathsf{T}\|_{\mathrm{otyp}} \leq \|\mathsf{T}\|_{\Pi^1_1} \leq \varepsilon_0.$$

But because we are not interested about this in this diploma thesis we only restrict ore self to $\|\mathsf{T}\|_{\Pi^1_1}$.

3.6 An upper bound for the Π^1_1-Ordinal of TA_f

We will give a constructive[13] proof of

$$\|\mathsf{TA}_f\|_{\Pi^1_1} \le \varepsilon_0.$$

Theorem 3.52.

$$\|\mathsf{TA}_f\|_{\Pi^1_1} \le \varepsilon_0$$

Proof. Assume φ is a Π^1_1-sentence of $\mathcal{L}_\mathsf{T}(\tau_{\mathrm{nt}})$ and $\mathsf{TA}_f \vdash \varphi$, then it is easy to see that $\mathsf{TA}_f \vdash \varphi^\triangledown$. From Corollary 3.42 we obtain that $\mathsf{NT}_\infty \vdash \iota(\varphi^\triangledown)[\alpha, 0]$ where $\alpha < \varepsilon_0$. Lemma 3.50 leads to $\mathbb{N} \models^\alpha \varphi^\triangledown$ where $\alpha < \varepsilon_0$. So we obtain $\mathrm{tc}(\varphi) < \varepsilon_0$. Since φ was arbitrary this holds for all these sentences. Since $\|\mathsf{TA}_f\|_{\Pi^1_1}$ is defined as a supremum (see Definition 3.51) the theorem is proved. □

Note by similar methods (without the use of infinite languages) we can also proof

$$\|\mathsf{TA}_f\|_{\mathrm{Con}} = \varepsilon_0$$

as Friedman showed it in [4] or Schwichtenberg showed in [13].

[13]Constructive means that the proof constructs an ordinal smaller then ε_0 for each φ with $\mathsf{TA}_f \vdash \varphi$ instead of only proving the existence of such an ordinal.

References

[1] H. Bachmann: *Transfinite Zahlen.* Springer-Verlag, second edition, Berlin 1967.

[2] H. B. Enderton: *Mathematical Introduction to Logic.* Harcourt/Academic Press, second edition, Burlington 2001.

[3] W. M. Farmer: *A unification-theoretic method for investigating the k-provability problem.* Annals of Pure and Applied Logic, 51, 1991, pp. 173-214.

[4] H. Friedman, M. Sheard: *Elementary descent recursion and proof theory.* Annals of Pure and Applied Logic, 71, 1995, pp. 1-45.

[5] G. Gentzen: *Die Widerspruchsfreiheit der reinen Zahlentheorie.* Mathematische Analen, 112, 1936, pp. 493-565.

[6] G. Gentzen: *Neue Fassung des Widerspruchsfreiheitsbeweises für die reine Zahlentheorie.* Forschungen zur Logik und zur Grundlegung der exakten Wissenschaften, Neue Reihe, 4, 1938, pp. 19-44.

[7] K. Kunen: *Set Theory, An Introduction to Independence Proofs.* North-Holland Publishing Co., tenth impression, Amsterdam 2006.

[8] W. Pohlers: *Proof Theory: The First Step into Impredicativity.* Springer-Verlag, second edition, Berlin 2009.

[9] M. Rathjen: *The Realm of Ordinal Analysis.* In: Set and Proofs, London Math. Soc. Lecture Note Ser. 258, edited by S. B. Cooper and J. K. Truss, Cambridge University Press, Cambridge 1999, pp. 219-279

[10] K. Schütte: *Beweistheoretische Erfassung der unendlichen Induktion in der Zahlentheorie.* Mathematische Annalen, vol. 122, 1955, pp. 369-389.

[11] K. Schütte: *Beweistheorie.* Springer-Verlag, Berlin 1960.

[12] K. Schütte: *Proof Theory.* Springer-Verlag, Berlin 1977.

[13] H. Schwichtenberg: *Proof Theory: Some Applications of Cut-Elimination.* In: Handbook of Mathematical Logic, edited by J. Barwise, North-Holland Publishing Co., Amsterdam 1977, pp. 867-895.

[14] C. Smorynski: *Proof Theory: The Incompleteness Theorem.* In: Handbook of Mathematical Logic, edited by J. Barwise, North-Holland Publishing Co., Amsterdam 1977, pp. 821-865.

[15] W. W. Tait: *Normal Derivability in Classical Logic.* In: The Syntax and Semantics of Infinitary Languages, edited by J. Barwise, Springer-Verlag, Berlin 1968, pp. 204-236.

[16] G. Takeuti: *Proof Theory*. North-Holland Publishing Co., second edition, Amsterdam 1987.

Abstract

In [6] Gerhard Gentzen prove the consistency of first-order Peano arithmetic **PA**. The method works as follows: Define a simple part **SPA** of peano arithmetic (**SPA** does in particular not contain induction) and first show the consistency of **SPA**. Now assume towards a contradiction that **PA** deducts an contradiction. Show that this deduction can be transformed into a deduction in **SPA**, this contradicts the consistency of **SPA**. How to get a deduction in **SPA**: We assign an ordinal (more exact an ordinal term of an ordinal notation system) to each deduction in **PA**, called the rank of the deduction. Next show that for each deduction which deducts a contradiction (and is not in **SPA**) there is a deduction (also deducting and contradiction) with smaller rank. This method requires that the ordinal notation system (which goes up to ε_0) is well-founded. It turns out that Gentzen's method requires only to the following properties of **PA**:

1. All axioms of **PA** are universal sentences or instances of the induction schema
$$\varphi(0) \wedge \forall x[\varphi(x) \to \varphi(x+1)] \to \forall x \varphi(x).$$
2. All closed terms are provable equal to a term build up just from 0 and the symbol of the successor function.

This allows a slight generalisation of Gentzen's method. In this Diploma Thesis we consider theories $T_f = (T_f)_0 \cup (T_f)_{\mathrm{Ind}}$ with the following properties:

1. $(T_f)_0$ contains only universal sentences.
2. $(T_f)_{\mathrm{Ind}}$ contains all instances of the general induction schema
$$\varphi(c_1) \wedge ... \wedge \varphi(c_m) \wedge \forall \vec{x}[\varphi(x_1) \wedge ... \wedge \varphi(x_n) \to \varphi(f(x_1, ..., x_n))] \to \forall x \varphi(x).$$
3. The simple part of T_f proves for every closed term t the equality of t to a term $\bar{t}$ build up just from $c_1, ..., c_m$ and f.

As in [6] for **PA**, the consistency of T_f can be shown with respect to their simple part which corresponds to the simple part of Gentzen (also without induction). As a consequence, one gets the following result for all such theories.

Corollary. *Assume $\varphi(a)$ is quantifier free and T_f consistent.*
If $T_f \models \exists x \varphi(x)$, then $(T_f)_0 \models \exists x \varphi(x)$.
I.e., T_f is Σ^0_1-conservative over $(T_f)_0$.

It seems that this method is different in an essential way to the method Kurt Schütte uses in his consistency proof of **PA**. Schütte, Tait and others uses calculi with infinite deduction rules. These methods compute, in some

sense, the proof theoretical ordinal of the considered theory by embedding the deductions of the theory (in ordinary first-order logic) in an infinite system which allows cut-elimination. In contrast to Gentzen's method Schütte's and Tait's methods are closely related to the proof theoretical ordinals.
We do not provide an analysis of the disparities of both methods. Instead we present the point of view Wolfram Pohlers take in [8], to measure the Π^1_1-ordinal of theories TA_f (presented as a Tait-calculus) satisfying the following conditions:

1. TA_f includes all defining axioms for primitive recursive functions.

2. All instances of the schema

   $$\varphi(c_1) \wedge ... \wedge \varphi(c_m) \wedge \forall \vec{x}[\varphi(x_1) \wedge ... \wedge \varphi(x_n) \rightarrow \varphi(f(x_1, ..., x_n))] \rightarrow \forall x \varphi(x)$$

   are included. Here f is an m-array primitive recursive function constant and $c_1, ..., c_l$ are individual constants.

3. Every $n \in \mathbb{N}$ is equal to a composition of $f^{\mathbb{N}}$ and the elements $c_1^{\mathbb{N}}, ..., c_l^{\mathbb{N}}$.

Abstract(German)

In [6] wird von Gerhard Gentzen die Wiederspruchsfreiheit der Peano Arithmetik erster Stufe **PA** bewiesen. Die Methode geht dabei follgendermaßen vor: Man definiert einen simplen Teil **SPA** der Peano Arithmetik (**SPA** enthällt im speziellen keine Anwedung des Induktionsschemas) und zeigt zuerst die Wiederspruchsfreiheit von **SPA**. Der Rest des Arguments verläuft indirekt. Man nimmt an, dass **PA** einen Wiederspruch ableitet und zeigt das dessen Deduktion zu einer Deduktion in **SPA** tranformiert werden kann, was der Wiederspruchsfreiheit von **SPA** wiederspricht. Diese Transformation verläuft wie folgt: Jeder Deduktion in **PA** wird eine Ordinalzahl (oder genauer, ein Ordinalzahlterm eines Ordinalzahlnotations Systems) zugeordnet, diese wird der Rang der Deduktion genannt. Dann wird gezeigt, dass es zu jeder Deduktion eines Wiederspruches (die nicht in **SPA** verläuft) eine Deduktion (ebenfalls eines Wiederspruches) gibt die einen kleineren Rang hat. Diese Methode benötigt daher die Wohlfundiertheit des verwendeten Ordinalzahlnotations Systems (in diesem fall bis ε_0). Bei näherer betrachtung von Gentzens Methode fällt auf, dass sie lediglich folgende Eigenschaften von **PA** verwendet:

1. Alle Axiome von **PA** sind Allsätze oder Instanzen des Induktionsschemas
$$\varphi(0) \wedge \forall x[\varphi(x) \to \varphi(x+1)] \to \forall x \varphi(x).$$

2. Alle geschlossenen Terme sind beweisbar (in **SPA**) gleich zu einem Term der lediglich aus 0 und dem Symbol der Nachfolgerfunktion aufgebaut ist.

Dies erlaubt eine Verallgemeinerung von Gentzens Methode. In dieser Diplomarbeit werden wir daher Theorien $T_f = (T_f)_0 \cup (T_f)_{\text{Ind}}$ betrachten die follgende Eigenschaften erfüllen:

1. $(T_f)_0$ besteht lediglich aus Allsätzen.

2. $(T_f)_{\text{Ind}}$ beinhaltet alle Instanzen des Induktionsschemas
$$\varphi(c_1) \wedge ... \wedge \varphi(c_m) \wedge \forall \vec{x}[\varphi(x_1) \wedge ... \wedge \varphi(x_n) \to \varphi(f(x_1, ..., x_n))] \to \forall x \varphi(x).$$

3. Der simple Teil von T_f beweist für jeden geschlossenen Term t, dass t gleich einem Term $\bar{t}$ ist der lediglich aus den symbolen $c_1, ..., c_m$ und f aufgebaut ist.

Die Wiederspruchsfreiheit von T_f kann nun, wie in [6] für **PA**, relativ zu ihrem simplen Teil (wo Induktion wie zuvor bei Gentzen nicht möglich ist) gezeigt werden. Eine konsequentz dieses Resultates ist das follgende Korollar.

Korollar. *Sei $\varphi(a)$ quantorenfrei und T_f wiederspruchsfrei.*
Wenn $T_f \models \exists x\varphi(x)$, dann $(T_f)_0 \models \exists x\varphi(x)$.
Insbesondere ist T_f Σ^0_1-konservativ über $(T_f)_0$.

Es scheint mir als wäre die Methode, die von Kurt Schütte in seinem Wiederspruchsfreiheitsbeweis von **PA** verwendet wird, eine gänzlich andere. Schütte, Tait und Andere verwenden Kalküle mit unendlichen Deduktionsregeln um, in einem gewissen Sinne, die Beweistheoretische Ordinalzahl einer Theorie zu berechnen. Dies erfolgt über eine Transformation der endlichen Deduktionen der Theorie (in der Logik erster Stufe) in Deduktionen in einem unendlichen Kalkül, das Schnittelimination erlaubt. Im Gegensatz zu Gentzens Methode hat die von Schütte eine enge Beziehung zu den beweistheoretischen Ordinalzahlen.
Auf die Unterschiede der beiden Methoden wird nicht weiter eingeangen werden. Anstatt dieses Vergleiches wird lediglich eine Variante von Taits Methode dazu verwendet die Π^1_1-Ordinalzahl, wie von Wolfram Pohlers in [8] beschrieben, von Theorien TA_f (aufgefasst als Taitkalkühl) zu messen. Es wird angenommen das TA_f follgende Eigenschaften erfüllt:

1. TA_f enthällt für jede primitiv rekursive Funktion die definierenden Formeln als Axiome.

2. Weiters enthällt TA_f alle Instanzen des Schemas

   $$\varphi(c_1)\wedge...\wedge\varphi(c_m)\wedge\forall\vec{x}[\varphi(x_1)\wedge...\wedge\varphi(x_n)\rightarrow\varphi(f(x_1,...,x_n))]\rightarrow\forall x\varphi(x).$$

   Hierbei ist f ein m-stelliges Symbol einer gleichstelligen primitiv rekursiven Funktion und $c_1,...,c_l$ Individuenkonstanten.

3. Es wird auserdem angenommen das jedes $n\in\mathbb{N}$ gleich einer Komposition aus $f^{\mathbb{N}}$ und den natürlichen Zahlen $c_1^{\mathbb{N}},...,c_l^{\mathbb{N}}$ ist.

Curriculum Vitae

Personal Data

Name	TOPPEL Michael
Date of birth	March 10, 1987
Nationality	Austria

Education

2007-2009	Faculty of philosophy and educational science, University of Vienna, studies in philosophy (discontinued)
2006-	Faculty of mathematics, University of Vienna, studies in mathematical logic, estimated finishing date: end of August 2011, Degree: Master (Mag. rer. nat.)
2001-2006	College for chemical engineering (HBLVA für Chemische Industrie - Rosensteingasse), Vienna, Austria

Professional Experience

2010-2011	Tutor(teaching assistant) at the institute of philosophy, University of Vienna
2005	Oncological research (internship) at Boehringer-Ingelheim Austria Gesmbh