A flexible type system for the small Veblen ordinal

Florian Ranzi* Thomas Strahm**

January 5, 2016

Abstract

We introduce and analyze two theories for typed (accessible part) inductive definitions and establish their proof-theoretic ordinal to be the small Veblen ordinal $\vartheta\Omega^\omega$. We investigate on the one hand the applicative theory FIT of functions, (accessible part) inductive definitions, and types. It includes a simple type structure and is a natural generalization of S. Feferman's system $\mathrm{QL}(\mathsf{F}_0\text{-}\mathsf{IR}_N)$. On the other hand, we investigate the arithmetical theory TID of typed (accessible part) inductive definitions, a natural subsystem of ID_1, and carry out a wellordering proof within TID. In particular, we present an ordinal notation system based on the finitary Veblen functions.

Keywords: Proof theory, inductive definitions, applicative theories, small Veblen ordinal, finitary Veblen functions, metapredicativity, higher types, subsystems of second-order arithmetic.

Contents

*Institut für Informatik und angewandte Mathematik, Universität Bern, Neubrückstrasse 10, CH-3012 Bern, Switzerland. Email: ranzi@inf.unibe.ch. Research supported by the Swiss National Science Foundation.

**Institut für Informatik und angewandte Mathematik, Universität Bern, Neubrückstrasse 10, CH-3012 Bern, Switzerland. Email: strahm@inf.unibe.ch. Homepage: http://www.inf.unibe.ch/~strahm.

1. Introduction

In [Fef92], S. Feferman introduced a two-sorted quantificational logic and showed that it has the same strength as (Skolem's system of) primitive recursive arithmetic. The characteristics of this two-sorted quantificational logic are that it is an *applicative theory* augmented by type variables as the second sort and with a refined notion of comprehension terms, so-called type and function terms. In particular, this theory embodies a rule (F_0-IR_N) that is called the function-induction rule on N (where N is interpreted as the type for the natural numbers). It was shown to be closed under a strengthening of this rule to *finitary* inductively generated types I, called (F_0-IR_I).

This kind of theory strongly influenced the shape of our applicative theory FIT that we are going to introduce in this article. Our motivation to examine the theory in [Fef92] was to find a natural theory for carrying out metapredicative[1] wellordering proofs in the spirit of higher type functionals for ordinals. It seemed to provide a suitable environment for doing so. But soon, we realized that aside from this, the theory gave rise to the question of what consequence a function-induction rule for *infinitary* inductively generated types would have on the one side and to the idea of implementing the wellordering proofs through accessible part inductive definitions on the other side (having in mind our desire for metapredicative wellordering proofs). Hence, we tackle this question on infinitary inductively generated types only for inductively generated types that correspond to the (inductively defined) *accessible part* $\mathsf{I}_{\mathbb{P},\mathbb{Q}}$ for a (binary) relation $\mathbb{Q}$ on a domain $\mathbb{P}$. In fact, our methods implicitly suggest that we get the same result for the variant where we allow for general inductively generated types.

FIT stands for "*theory for function(al)s, non-iterated inductive definitions, and types (of level 1)*", and it represents the first step for a generalization of the theory in [Fef92] which turns out to have the small Veblen ordinal as measure for its proof-theoretic strength, i.e., $\vartheta\Omega^\omega$ when

[1]The notion *metapredicativity* is meant in general for the approach to use proof-theoretic methods form the realm of *predicative* proof-theory instead of *impredicative* methods. In particular for wellordering proofs, we aim to avoid the use of so-called collapsing functions. We refer to [Str99] or [JKSS99]. For further reading on *metapredicativity*, we refer to [Jäg05] and [JS05].

using the terminology of [RW93]. Theories that have $\vartheta\Omega^\omega$ as proof-theoretic strength are for instance $\Pi^1_2\text{-}\mathsf{BI}_0$ from [RW93] or more recently $\mathsf{RCA}_0+(\Pi^1_1(\Pi^0_3)\text{-}\mathsf{CA}_0)^-$ from [vdMRW14]. While these theories are analyzed by *impredicative* proof-theoretic methods, our treatment of FIT uses *metapredicative* methods for the lower bound. For the upper bound, we use an embedding into $\Pi^1_3\text{-}\mathsf{RFN}_0$ and get a desired upper bound result in the realm of metapredicative proof-theory due to D. Probst's *modular ordinal analysis* from [Pro16] that determines by metapredicative methods the proof-theoretic ordinal of various theories with strength below (and reaching to) the Bachmann-Howard ordinal $\vartheta\varepsilon_{\Omega+1}$. One of these theories is $\Pi^1_3\text{-}\mathsf{RFN}_0$ (which is denoted by $\mathsf{p}_3(\mathsf{ACA}_0)$ in [Pro16]) and determined to have the proof-theoretic strength of the small Veblen ordinal. Furthermore, we mention the system $\mathsf{KPi}^0+(\Pi_3\text{-}\mathsf{Ref})$ from [JS05] which is also related to $\Pi^1_3\text{-}\mathsf{RFN}_0$. In particular, [JS05] explains how the proof-theoretic strength of $\mathsf{KPi}^0+(\Pi_3\text{-}\mathsf{Ref})$ can be determined to be $\vartheta\Omega^\omega$ by metapredicative methods.

Results on the Theory FIT

We now explain the methods used in this article for the ordinal analysis of FIT. First, we shall consider a canonical implementation of FIT as a subsystem of ID_1 (while the latter is an arithmetical first-order theory for non-iterated general inductive definitions, see [BFPS81]) in which metapredicative wellordering proofs can be carried out in a perspicuous way and where the interpretation back into FIT is straight-forward. This subsystem of ID_1 is called TID for "*theory of typed (accessible part) inductive definitions (of level* 1*)*" and essentially arises from ID_1 by restricting to accessible part inductive definitions and adapting the closure axioms, its induction scheme on the natural numbers, and its generalized induction scheme (ID) to (the translation of) the function types of FIT, akin to the restriction of ID_1 to the theory $\mathsf{ID}^*_1\!\restriction$ from [Pro06].

For the (proof-theoretic) upper bound of FIT (and hence for TID), we shall embed it into the system $\Pi^1_3\text{-}\mathsf{RFN}_0$ of second-order arithmetic for Π^1_3 ω-model reflection. In order to obtain the desired upper bound $\vartheta\Omega^\omega$, we shall use the results from [RW93] by impredicative methods, noting that the (meta)predicative treatment from [Pro16] has not been published yet. Figure 1 depicts the abovementioned approaches accordingly. Furthermore, we give in the conclusion of this article (Section 8) some remarks on the canonical generalization of TID to a theory $\mathsf{TID}^{\mathbf{f}}$ for general typed inductive definitions with the *full* range of positive arithmetical operator forms, leading to the same proof-theoretic strength of TID and $\mathsf{TID}^{\mathbf{f}}$. We refer here also to [Ran15] for an extension TID^+_1 of TID with proof-theoretic ordinal $\vartheta\Omega^\Omega$, i.e., the large Veblen ordinal.

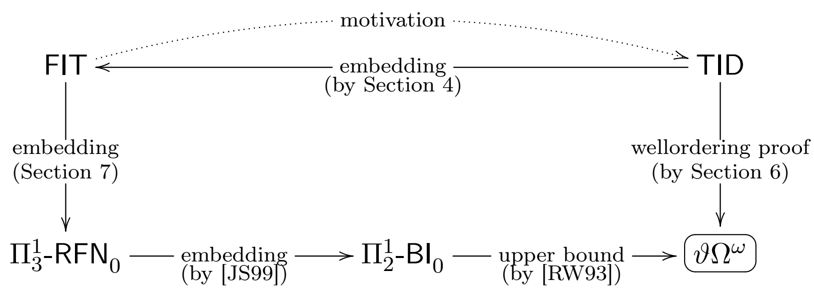

Figure 1: Strategy to determine the proof-theoretic ordinal $\vartheta\Omega^\omega$ of FIT

2. General Notational Framework

In this article, we shall work with three conceptually different kinds of logical frameworks: First in Section 3 with the two-sorted theory FIT (with language $\mathcal{L}_{\mathsf{FIT}}$) that is an applicative theory enhanced by a type system, then in Section 4 with the first-order theory TID (with language $\mathcal{L}_{\mathsf{TID}}$) that is an extension of Peano arithmetic PA (with language $\mathcal{L}_{\mathsf{PA}}$) by new predicates, and eventually in Section 7 with some subsystems of second order arithmetic (with language $\mathcal{L}^2_{\mathsf{PA}}$). Hence, we work in this article with up to two sorts of (countably many) variables and we use

$$\begin{array}{ll} a, b, c, d, u, v, w, x, y, z & \text{as syntactic variables for the first sort,} \\ U, V, W, X, Y, Z & \text{as syntactic variables for the second sort,} \end{array}$$

and choose

$$=, \neg, \rightarrow, \vee, \wedge, \exists, \forall \qquad \text{as basic logical symbols.}$$

Now, let $\mathcal{L}$ be one of the languages $\mathcal{L}_{\mathsf{FIT}}$, $\mathcal{L}_{\mathsf{TID}}$, $\mathcal{L}_{\mathsf{PA}}$, or $\mathcal{L}^2_{\mathsf{PA}}$, and assume that the notion of $\mathcal{L}$ terms and $\mathcal{L}$ formulas has been already introduced. In case that $\mathcal{L}$ is clear from the context, we shall sometimes drop the reference to $\mathcal{L}$ by just using the notions term and formula.

- s, t, r shall be used as syntactic variables for $\mathcal{L}$ terms.
- A, B, C, D, E, F shall be used as syntactic variables to denote $\mathcal{L}$ formulas, and we call an atomic $\mathcal{L}$ formula or its negated version a *literal*.
- If an $\mathcal{L}$ formula is introduced as $A(a)$, this means that A denotes this formula and that the variable a may occur freely in A (i.e., a is not in the scope of any $\forall a$ or $\exists a$ quantification).
- $\mathrm{FV}(A)$ denotes the set of free variables of the first sort of A.
- a, b, c, d, u, v, w, shall primarily be used within an $\mathcal{L}$ formula to denote free variables of the first sort.
- k, l, m, n, p, q shall primarily be used as variables in our meta-theory, i.e., as ranging there over the natural numbers.
- Parentheses may be added or dropped in order to make expressions unambiguous or more readable, e.g., we may write a quantification in the form $\exists x A$, $(\exists x) A$, or $\exists x(A)$.
- We often prefer *infix notation* rather than prefix notation when dealing with binary function and relation symbols.
- For $\rightarrow$, we follow the usual convention of right-associativity, e.g., $A \rightarrow B \rightarrow C$ denotes $A \rightarrow (B \rightarrow C)$. We further write $A \leftrightarrow B$ to denote $(A \rightarrow B) \wedge (B \rightarrow A)$. Moreover, $\wedge$ binds stronger than $\rightarrow$.

Furthermore, we settle the following notational conventions.

Vector Notations

If $*$ denotes one of the syntactic variables that will be introduced in this article, then we allow the usual annotations such as $*'$, $\tilde{*}$, or subscripts $*_i$ (for $i \in \mathbb{N}$, i.e., for natural numbers i). With respect to subscripts, we also use the vector notation $\vec{*}$ to denote lists of the form $*_1, \ldots, *_n$ for some $n \in \mathbb{N}$. If we introduce a list as $\bar{*}^{(n)}$ for a particular $n \in \mathbb{N}$ and a syntactic variable $*$, then we mean $*_1, \ldots, *_n$, and we may write $\bar{*}^{(k)}$ for any $k \in \mathbb{N}$ in order to denote $*_1, \ldots, *_{\min(k,n)}$. In some rare cases we may write for specific constants c (e.g., for 0) the expression $\bar{c}^{(n)}$ to denote the list $c, \ldots, c$ of length n, and hence we read $\vec{c}$ analogously. This notation will come in handy in particular when we will be working with ordinal notations that are based on the finitary Veblen functions. If $n = 0$, then $\bar{*}^{(n)}$ and $\vec{*}$ denote the empty list.

Applications of all these notations will be obvious, following common conventions—for instance $\forall \bar{x}^{(3)} A$ shall abbreviate $\forall x_1 \forall x_2 \forall x_3 A$ as usual, and $\forall \vec{x} A$ is just A if $\vec{x}$ is the empty list. Also when writing $f\bar{t}^{(n)}$ for a list of terms $\bar{t}^{(n)}$ and an n-ary function symbol f, it is usually meant to abbreviate $ft_1 \ldots t_n$ rather than $ft_1, \ldots, t_n$.

Class Terms and Substitution

$\mathcal{L}$ *class terms* are objects of the form

$$\Lambda a.A$$

for any $\mathcal{L}$ formula A and we use $\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}$ as syntactic variables for $\mathcal{L}$ class terms. Sometimes, class terms are also called comprehension terms, and we do not use the more common notations $\{a \colon A\}$ or $\lambda a.A$ because these notions are already reserved in our setting of the applicative theory FIT.

Substitution of a variable a in an $\mathcal{L}$ formula A by an $\mathcal{L}$ term t is denoted by $A(t/a)$ and $A_a(t)$, or just by $A(t)$ in case A has been introduced in the form $A(a)$, and as usual we assume (if necessary) an appropriate renaming of bound variables in A to avoid a clash of bound variables. Then for $\mathcal{A}$ being $\Lambda a.A$, we set

$$\mathcal{A}(t) := t \in \mathcal{A} := A(t/a)$$

for any $\mathcal{L}$ term t and we ambiguously write $\mathcal{A} \in \mathcal{L}$ to stress that $\mathcal{A}$ is an $\mathcal{L}$ class term. Moreover, we also extend this to lists of variables $\vec{a} = a_1, \ldots, a_n$ and have objects of the form

$$\Lambda a_1. \ldots \Lambda a_n.A$$

or $\Lambda \vec{a}.A$ for short with $(\Lambda \vec{a}.A)(\vec{t}) := A(\vec{t}/\vec{a})$ for terms $\vec{t} = t_1, \ldots, t_n$ and where $A(\vec{t}/\vec{a})$ is obtained by simultaneously replacing in A all free occurrences of the variables $\vec{a}$ by $\vec{t}$, while renaming bound variables may be necessary to avoid a clash of variables. We use $(\Lambda \bar{a}^{(n)}.A)(\bar{t}^{(k)})$ for $k < n$ to denote $(\Lambda a'_{k+1}. \ldots \Lambda a'_n.\big(\Lambda \bar{a}^{(k)}.(A(a'_{k+1}, \ldots, a'_n / a_{k+1}, \ldots, a_n))(\bar{t}^{(k)})\big) \ldots)$ where $a'_{k+1}, \ldots, a'_n$ are fresh variables that do not appear in $\bar{t}^{(k)}$, $\bar{a}^{(n)}$, A.

In case that $\mathcal{L}$ also embodies variables X, Y, Z of the second sort, we mean by *substitution of a variable* X in an $\mathcal{L}$ formula A by an $\mathcal{L}$ class term $\mathcal{B}$ the expression

$$A(\mathcal{B}/X)$$

which is obtained from A by substituting any atomic formula Xt with $\mathcal{B}(t)$ while a renaming of bound variables may be necessary as usual. If A has been introduced in the form $A(X)$, we may also just write $A(\mathcal{B})$ for $A(\mathcal{B}/X)$.

In case R is a unary relation symbol in $\mathcal{L}$ or a second sort variable, we also define

$$A(R/X) := A(\mathcal{B}/X)$$

for $\mathcal{B} := \Lambda a.Ra$. Furthermore, if $\mathcal{A}$ is an $\mathcal{L}$ class term $\Lambda a.A$, then we set

$$\mathcal{A}(\mathcal{B}/X) := \Lambda a.A(\mathcal{B}/X)$$

Accordingly, we let substitution for number variables be defined by $\mathcal{A}_z(t) := \mathcal{A}(t/z) := \Lambda a.A(t/z)$ if a does not occur in t, and otherwise we let $\mathcal{A}_z(t) := \mathcal{A}'_z(t)$ for $\mathcal{A}' := \Lambda b.A(b/a)$ and some b that does not occur in A, t.

3. The Theory FIT for Functions, Inductive Definitions, and Types

The full language of FIT will be defined in Subsection 3.1. Here, we shall introduce a basic language that is needed for the applicative part of FIT.

Definition 3.1. The *basic language* of FIT is built-up on two sorts of variables, while first sort variables are called *individual variables* and second sort variables are called *type variables*. The basic language further consists of the following symbols.

(a) *Constants of the first sort:*

$\mathsf{k}, \mathsf{s}, \mathsf{p}, \mathsf{p}_0, \mathsf{p}_1, 0, \mathsf{s_N}, \mathsf{p_N}, \mathsf{d_N}$ (denoting the usual applicative constants)

(b) *Constants of the second sort:*

N (denoting the natural numbers)

$\overline{\mathsf{N}}$ (denoting the complement of the natural numbers)

U (without further interpretation and needed for proof-theoretic investigations)

(c) *Relation symbols of the first sort:*

$=$ (denoting equality on individual terms[2])

$\downarrow$ (denoting definedness for individual terms)

(d) *Further symbols:*

$\cdot$ (denoting a binary function symbol for first sort term application)

$\in$ (denoting a binary relation symbol between individual terms and types[3])

Definition 3.2. *Individual terms* s, t, r are defined inductively from individual variables and constants by use of the binary function symbol $\cdot$ as usual.

Definition 3.3. The following notions and abbreviations will serve as basic applicative tools.

(a) $t' := \mathsf{s_N} t$ and $1 := 0'$.

[2]Individual terms will be defined in Definition 3.2.
[3]Types will be defined in Definition 3.5.

(b) *Term application on n inputs* is defined recursively on $n \geq 0$:

$$st_1 \dots t_n := s(t_1, \dots, t_n) := \begin{cases} s & \text{if } n = 0 \\ (s \cdot t_1)t_2 \dots t_n & \text{if } n > 0 \end{cases}$$

(c) *General n-tupling* is defined recursively on $n \geq 0$:

$$\langle s_0, \dots, s_{n-1} \rangle^{\mathsf{FIT}} := \begin{cases} 0 & \text{if } n = 0 \\ \mathsf{p} s_0 \langle s_1, \dots, s_{n-1} \rangle^{\mathsf{FIT}} & \text{if } n > 0 \end{cases}$$

Write shall write $\langle s_0, \dots, s_{n-1} \rangle$ for $\langle s_0, \dots, s_{n-1} \rangle^{\mathsf{FIT}}$ if the meaning is clear from the context.

(d) The *n-th projection* is defined recursively on $n \geq 0$:

$$(s)_n^{\mathsf{FIT}} := \begin{cases} \mathsf{p}_0 s & \text{if } n = 0 \\ (\mathsf{p}_1 s)_{n-1}^{\mathsf{FIT}} & \text{if } n > 0 \end{cases}$$

Write shall write $(s)_n$ for $(s)_n^{\mathsf{FIT}}$ if the meaning is clear from the context.

(e) *Lambda abstraction of a variable x* on a term t is defined recursively on the build-up of t:

$$\lambda x.t := \begin{cases} \mathsf{skk} & \text{if } t \text{ is } x \\ \mathsf{k}t & \text{if } t \text{ is a constant or a variable different from } x \\ \mathsf{s}(\lambda x.t_1)(\lambda x.t_2) & \text{if } t \text{ is } t_1 t_2 \end{cases}$$

while note that $\lambda x.t$ does not contain the variable x. In general, *lambda abstraction of a list of variables $\vec{x} = x_1, \dots, x_n$* over a term t is defined recursively on $n \geq 0$:

$$\lambda \vec{x}.t := \begin{cases} t & \text{if } n = 0 \\ \lambda x_1.(\lambda x_2 \dots x_n.t) & \text{if } n > 0 \end{cases}$$

Remark 3.4. $\langle\rangle$ appears for instance in the proof of Lemma 4.29.

3.1. Syntax of FIT

Definition 3.5. The language $\mathcal{L}_{\mathsf{FIT}}$ is defined simultaneously and inductively with the notions for *formulas* (For), *positive formulas* (For^+), *types* (Ty), *restricted types* ($\text{Ty}{\restriction}$), and *terms of the second sort*:

(a) $\mathcal{L}_{\mathsf{FIT}}$ extends the basic language from Definition 3.1 by new (syntactically different) kinds of *terms of the second sort*

$$\{x \colon A\} \qquad \text{and} \qquad \mathsf{I}_{\mathbb{P},\mathbb{Q}}$$

demanding here $A \in \text{For}^+$ and $\mathbb{P}, \mathbb{Q} \in \text{Ty}{\restriction}$.

(b) For denotes the collection of *formulas* A, B, C, D, which consists of the expressions

$$t \in \mathbb{P} \quad t \in \mathsf{U} \quad t\downarrow \quad s = t$$
$$\neg A \quad A \to B \quad A \vee B \quad A \wedge B \quad \exists x A \quad \forall x A \quad \exists X A \quad \forall X A$$

and we demand here $\mathbb{P} \in \mathrm{Ty}$. We sometimes write $A \in \mathcal{L}_{\mathsf{FIT}}$ ambiguously for $A \in \mathrm{For}$.

(c) For^+ denotes the collection of *positive (elementary) formulas*, i.e., formulas $A \in \mathrm{For}$ such that

- quantifications of type variables do not occur and
- expressions of the form $t \in \mathbb{P}$ for types $\mathbb{P} \in \mathrm{Ty}$ occur at most positively[4]

(d) Ty denotes the collection of *types* $\mathbb{P}, \mathbb{Q}, \mathbb{R}$ (also called *positive types*), i.e., expressions of the form

$$X, Y, Z, \ldots \text{ (i.e., type variables)}$$
$$\mathsf{N} \quad \overline{\mathsf{N}} \quad \{x \colon A\} \quad \mathsf{I}_{\mathbb{P},\mathbb{Q}}$$

demanding here $A \in \mathrm{For}^+$ and $\mathbb{P}, \mathbb{Q} \in \mathrm{Ty}{\restriction}$. Note that U itself is not treated as a type.

(e) $\mathrm{Ty}{\restriction}$ denotes the collection of *restricted types*, i.e., types such that

- no type variables and
- no expressions of the form $\mathsf{I}_{\mathbb{P},\mathbb{Q}}$ occur

Definition 3.6. Let $\twoheadrightarrow$ be a new distinguished symbol. The collection FT of *function types* $\mathbb{F}, \mathbb{G}, \mathbb{H}$ is defined inductively to consist of expressions of the form

$$\mathbb{P} \qquad \text{and} \qquad \mathbb{P} \twoheadrightarrow \mathbb{F}$$

for any $\mathbb{P} \in \mathrm{Ty}$ and $\mathbb{F} \in \mathrm{FT}$. Note that function types are defined as objects in the meta-language.

We can write any $\mathbb{F} \in \mathrm{FT}$ in the form $(\mathbb{P}_1 \twoheadrightarrow (\ldots(\mathbb{P}_{n-1} \twoheadrightarrow \mathbb{P}_n)\ldots))$, and we allow to simplify this notation to $\mathbb{P}_1 \twoheadrightarrow \ldots \mathbb{P}_{n-1} \twoheadrightarrow \mathbb{P}_n$ by following the convention of right-associativity for $\twoheadrightarrow$.

Remark 3.7. We did not define U to be a type because we can use $\{x \colon x \in \mathsf{U}\}$ in FIT in order to get $t \in \mathsf{U}$, namely by making use of (CA^+) from Definition 3.12.

Definition 3.8 (Free variables and substitution)**.** The notion of $\mathrm{FV}(A)$ is extended to the notion of atomic formulas $t \in \mathbb{P}$ for $\mathbb{P} \in \mathrm{Ty}$ by defining recursively on the build-up of types and formulas:

- $\mathrm{FV}(t \in \mathbb{P}) := \mathrm{FV}(t) \cup \mathrm{FV}(\mathbb{P})$ and
- $\mathrm{FV}(\mathbb{P}) := \begin{cases} \mathrm{FV}(A) \setminus \{x\} & \text{if } \mathbb{P} \text{ is } \{x \colon A\} \\ \mathrm{FV}(\mathbb{P}') \cup \mathrm{FV}(\mathbb{Q}') & \text{if } \mathbb{P} \text{ is } \mathsf{I}_{\mathbb{P}',\mathbb{Q}'} \\ \emptyset & \text{otherwise} \end{cases}$

[4]Positive is meant in the usual way: $t \in \mathbb{P}$ is called *positive* in $A \in \mathrm{For}$ if it does not occur in negated form $\neg(t \in \mathbb{P})$ in A', while A' shall be the translation of A where first each subformula of the form $B_1 \to B_2$ is transformed to $\neg B_1 \vee B_2$ and where we then move the negation symbol $\neg$ next to atomic formulas, while making use of De Morgan's laws and the law of double negation.

With this extension explained, the *substitution of individual and type variables* is defined as in Section 2.

Notation 3.9. We have the following abbreviations for some formulas and types:

- $s \simeq t$ is $(s\downarrow \vee t\downarrow) \to s = t$.
- $s \neq t$ is $s\downarrow \wedge t\downarrow \wedge \neg(s = t)$.
- $t \in \mathbb{P} \twoheadrightarrow \mathbb{F}$ is recursively $\forall x(x \in \mathbb{P} \to tx \in \mathbb{F})$.
- $\mathsf{N}^{n+1} \twoheadrightarrow \mathbb{F}$ is recursively $\mathsf{N} \twoheadrightarrow (\mathsf{N}^n \twoheadrightarrow \mathbb{F})$ where $\mathsf{N}^0 \twoheadrightarrow \mathbb{F}$ is $\mathbb{F}$.
- $t \notin \mathbb{F}$ is $\neg(t \in \mathbb{F})$.
- $(\exists x \in \mathbb{F})B$ is $\exists x(x \in \mathbb{F} \wedge B)$.
- $(\forall x \in \mathbb{F})B$ is $\forall x(x \in \mathbb{F} \to B)$.
- $\mathrm{Cl}_{\mathbb{P},\mathbb{Q}}(\mathcal{A})$ is $\forall x\big((x \in \mathbb{P} \wedge (\forall y \in \mathbb{P})(\langle y, x\rangle \in \mathbb{Q} \to \mathcal{A}(y))\big) \to \mathcal{A}(x))$.

  We assume as usual for such notational abbreviations that x, y are supposed to not occur in $\mathcal{A}$, $\mathbb{P}$, and $\mathbb{Q}$. This shall hold analogously for similar such abbreviations for formulas.
- $A(\mathbb{F}/X)$ for the formula obtained by substituting any occurrence of $t \in X$ in A by $t \in \mathbb{F}$.

Remark 3.10.

(a) We chose $\mathrm{Cl}_{\mathbb{P},\mathbb{Q}}(\mathcal{A})$ to be defined with a conjunction rather than a chain of implications such as in $\forall x\big(x \in \mathbb{P} \to (\forall y \in \mathbb{P})(\langle y, x\rangle \in \mathbb{Q} \to \mathcal{A}(y)) \to \mathcal{A}(x)\big)$ which is logically equivalent to $\mathrm{Cl}_{\mathbb{P},\mathbb{Q}}(\mathcal{A})$. The reason for this is of syntactical nature, allowing for a simplified representation in Section 7 (cf., Remark 7.20).

(b) Note that function types are not necessarily part of the language $\mathcal{L}_{\mathsf{FIT}}$: We defined expressions of the form $\mathbb{P} \twoheadrightarrow \mathbb{F}$ from outside and in our meta-language, using the delimiter $\twoheadrightarrow$. Within $\mathcal{L}_{\mathsf{FIT}}$ formulas, these new expressions will only occur in the form $t \in \mathbb{P} \twoheadrightarrow \mathbb{F}$, i.e., as $\mathcal{L}_{\mathsf{FIT}}$ formulas.

    Alternatively and in order to make function types first-class members of $\mathcal{L}_{\mathsf{FIT}}$, we could have introduced a more general form of type (called *general type* as in [Fef92]), allowing for expressions $\{x\colon A\}$ for any $A \in \mathrm{For}$ and thus abbreviate $\mathbb{P} \twoheadrightarrow \mathbb{F}$ by $\{x\colon (\forall y \in \mathbb{P})(xy \in \mathbb{F})\}$ where x, y are any distinct individual variables that do not occur in $\mathbb{P}$ or $\mathbb{F}$, and then we would need to strengthen the comprehension scheme to allow for general types. This alternative approach does not change anything in the result because the comprehension scheme can be reduced to the variant we have in this article (this has been also done in [Fef92]).

(c) We used the restriction to $\mathrm{Ty}{\restriction}$ in the definition of $\mathsf{I}_{\mathbb{P},\mathbb{Q}} \in \mathrm{Ty}$ in order to account for a non-iterated inductive definition.

3.2. The Theory FIT

Definition 3.11. The logic of FIT is a two-sorted logic whose first-order part (i.e., for individual variables) is based on the *classical logic of partial terms* LPT due to Beeson [Bee85]:

- **Propositional axioms and rules.** The usual propositional axioms and rules, based on some sound Hilbert calculus for classical propositional logic.
- **Quantificational logic for the first sort.** For A being an $\mathcal{L}_{\mathsf{FIT}}$ formula and t an individual term, we have

$$(\forall x A \land t{\downarrow}) \to A(t/x)$$
$$(A(t/x) \land t{\downarrow}) \to \exists x A$$

  and for A, B being $\mathcal{L}_{\mathsf{FIT}}$ formulas and $x \notin \mathrm{FV}(A)$, we have the following figures:

$$\frac{A \to B}{A \to \forall x B} \qquad \frac{B \to A}{\exists x B \to A}$$

- **Quantificational logic for the second sort.** For A, B being $\mathcal{L}_{\mathsf{FIT}}$ formulas and $\mathbb{P}$ a type, we have

$$\forall X A \to A(\mathbb{P}/X)$$
$$A(\mathbb{P}/X) \to \exists X A$$

  and for A, B being $\mathcal{L}_{\mathsf{FIT}}$ formulas and X not occurring free in A, we have the following figures:

$$\frac{A \to B}{A \to \forall X B} \qquad \frac{B \to A}{\exists X B \to A}$$

- **Equality axioms.**

$$x = x$$
$$(x_1 = y_1 \land \ldots \land x_n = y_n \land A) \to (\ldots(A(y_1/x_1))\ldots(y_n/x_n))$$

- **Definedness axioms.** For all constants c of the first sort of $\mathcal{L}_{\mathsf{FIT}}$, we have

$$c{\downarrow} \land x{\downarrow}$$
$$(st){\downarrow} \to (s{\downarrow} \land t{\downarrow})$$
$$s = t \to (s{\downarrow} \land t{\downarrow})$$

  and for every type $\mathbb{P}$ and individual term t, we have

$$t \in \mathbb{P} \to t{\downarrow}$$
$$t \in \mathsf{U} \to t{\downarrow}$$

Writing $\vdash A$ for any $\mathcal{L}_{\mathsf{FIT}}$ formula A denotes the derivability of A in the logic of FIT.

Definition 3.12. FIT is the two-sorted applicative theory based on the logic of partial terms LPT (and on [Fef92]). Its non-logical axioms are as follows:

I. Applicative axioms.
I.1. Partial combinatory algebra.

$$\mathsf{k}xy = x$$
$$\mathsf{s}xy{\downarrow} \wedge \mathsf{s}xyz \simeq (xz)(yz)$$

I.2. Pairing and projection.

$$\mathsf{p}_0(\mathsf{p}xy) = x \wedge \mathsf{p}_1(\mathsf{p}xy) = y$$

I.3. Definition by numerical cases.

$$x \in \mathsf{N} \wedge y \in \mathsf{N} \wedge x = y \to \mathsf{d_N} z_1 z_2 xy = z_1$$
$$x \in \mathsf{N} \wedge y \in \mathsf{N} \wedge x \neq y \to \mathsf{d_N} z_1 z_2 xy = z_2$$

I.4. Axioms about N and $\overline{\mathsf{N}}$.

$$0 \in \mathsf{N} \wedge (x \in \mathsf{N} \to x' \in \mathsf{N})$$
$$x \in \mathsf{N} \to (x' \neq 0 \wedge \mathsf{p_N}(x') = x)$$
$$(x \in \mathsf{N} \wedge x \neq 0) \to (\mathsf{p_N}x \in \mathsf{N} \wedge (\mathsf{p_N}x)' = x)$$
$$x \in \overline{\mathsf{N}} \leftrightarrow x \notin \mathsf{N}$$

II. Induction on N for $\mathbb{F} \in \mathrm{FT}$.

(FT-Ind) $\quad t0 \in \mathbb{F} \wedge (\forall x \in \mathsf{N})(tx \in \mathbb{F} \to tx' \in \mathbb{F}) \to t \in (\mathsf{N} \twoheadrightarrow \mathbb{F})$

III. Positive comprehension for $A \in \mathrm{For}^+$.

(CA^+) $\quad y \in \{x \colon A\} \leftrightarrow A(y/x)$

IV. Axioms about $\mathsf{I}_{\mathbb{P},\mathbb{Q}}$ for $\mathbb{F} \in \mathrm{FT}$ and $\mathbb{P}, \mathbb{Q} \in \mathrm{Ty}{\restriction}$.

(FT-Cl) $\quad \mathrm{Cl}_{\mathbb{P},\mathbb{Q}}(\Lambda z.z \in \mathsf{I}_{\mathbb{P},\mathbb{Q}})$

(FT-ID) $\quad \mathrm{Cl}_{\mathbb{P},\mathbb{Q}}(\Lambda z.tz \in \mathbb{F}) \to t \in (\mathsf{I}_{\mathbb{P},\mathbb{Q}} \twoheadrightarrow \mathbb{F})$

Writing $\mathsf{FIT} \vdash A$ for any $\mathcal{L}_{\mathsf{FIT}}$ formula A denotes the derivability of A from these axioms in the logic of FIT given in Definition 3.11.

Lemma 3.13 (Basic applicative tools)**.**

(a) Lambda abstraction: For all $\mathcal{L}_{\mathsf{FIT}}$ terms t, s and $\vec{s} = s_1, \ldots, s_n$, and all individual variables y and $\vec{x} = x_1, \ldots, x_n$ with $y \notin \{x_1, \ldots, x_n\}$, we have the following:

1. $\mathsf{FIT} \vdash (\lambda\vec{x}.t){\downarrow} \wedge (\lambda\vec{x}.t)\vec{x} \simeq t$.
2. $\mathsf{FIT} \vdash (s_1{\downarrow} \wedge \ldots \wedge s_n{\downarrow}) \to (\lambda\vec{x}.t)\vec{s} \simeq t(\vec{s}/\vec{x})$.
3. $\mathsf{FIT} \vdash (\lambda\vec{x}.t)(s/y)x \simeq (\lambda\vec{x}.t(s/y))x$.

(b) Fixed-point: There exists a closed term fix *such that* $\mathsf{FIT} \vdash \mathsf{fix}y{\downarrow} \wedge \mathsf{fix}yx \simeq y(\mathsf{fix}y)x$ *holds for all number variables x, y.*

(c) Pairs and tupling: For all $\mathcal{L}_{\mathsf{FIT}}$ variables $x_0, \ldots, x_n$ and each $0 \leq i \leq n$, we have $\mathsf{FIT} \vdash (s_0{\downarrow} \wedge \ldots \wedge s_n{\downarrow}) \to (\langle s_0, \ldots, s_n \rangle)_i = s_i$.

Proof. The applicative part of FIT corresponds to the standard axioms and constants that appear in applicative theories. For details on (a) and (b), we refer to [FJS]. In particular for (c), we remark that this follows easily from the axioms of FIT, in particular by making use of **I.2.** from Definition 3.12 and the definedness axioms. □

3.3. Informal Interpretation of FIT

Since FIT directly evolved from Feferman's theory $\mathrm{QL}(\mathsf{F}_0\text{-}\mathsf{IR}_N)$, we refer for a thorough motivation and informal interpretation of FIT to [Fef92, Sections 2 and 5]. Moreover, the special constant U can be interpreted as a subset of the natural numbers, having no further interpretation. It is needed for proof-theoretic investigations.

4. The Theory TID for Typed Inductive Definitions

FIT is a natural theory for specifying the behaviour of an applicative term t by use of types, say by a function type $\mathbb{P}_1 \twoheadrightarrow \ldots \twoheadrightarrow \mathbb{P}_{n+1}$ that conists of types. For checking this behaviour, we have the axiom schemes (FT-Ind) and (FT-ID) at hand. The latter allows the discussion of the behaviour of an operation t that acts on the inductively defined accessible part of a given binary relation (e.g., if $\mathbb{P}_1$ is $\mathsf{I}_{\mathbb{P},\mathbb{Q}}$ in the example above). This gives an idea for the following definition of the theory TID for typed inductive definitions as a subtheory of ID_1. Before turning to the definition of TID, we introduce basic notions that we are going to use in combination with arithmetical theories.

Definition 4.1. $\mathcal{L}_{\mathsf{PA}}$ is the first-order language of Peano arithmetic with

- just one sort of variables x, referred to as *(number) variables*,
- a unary relation symbol U (without further interpretation and that is needed for proof-theoretic investigations),
- a symbol $=$ for equality,
- function symbols for each primitive recursive function, and for $n \in \mathbb{N}$, we denote by PR^n the collection of those function symbols that have arity n, and
- relation symbols R_f for each function symbol $f \in \mathrm{PR}^n$ with $n \neq 0$, and R_f has the same arity as f.

For the sake of completeness, we provide $\mathrm{PR} := \bigcup_{n\in\mathbb{N}} \mathrm{PR}^n$ via one of the usual formulations by an inductive definition over $n \in \mathbb{N}$ of function symbols, while $\mathbf{0}^n, \mathbf{S}, \mathbf{I}_i^{n+1}$ denote here symbols for the *constant zero function*, the *successor function*, and the *i-th projection function on $n+1$-tuple*, respectively, while $\mathbf{C}, \mathbf{R}$ are auxiliary symbols in our meta-theory for expressing *composition* and *primitive recursion*, respectively:

- $\mathbf{0}^n \in \mathrm{PR}^n$, $\mathbf{S} \in \mathrm{PR}^1$, and $\mathbf{I}_i^{n+1} \in \mathrm{PR}^{n+1}$ for each i with $1 \leq i \leq n+1$.
- $(\mathbf{C} f g_1 \ldots g_m) \in \mathrm{PR}^n$ if $f \in \mathrm{PR}^m$, $g_1, \ldots, g_m \in \mathrm{PR}^n$, and $m, n \geq 1$.
- $(\mathbf{R} f g) \in \mathrm{PR}^{n+1}$ if $f \in \mathrm{PR}^n$ and $g \in \mathrm{PR}^{n+2}$.

Remark 4.2. We added relation symbols R_f to $\mathcal{L}_{\mathsf{PA}}$ for technical reasons, namely in order to ease the embedding from TID into FIT in Section 4.2 (cf., Remark 4.26).

Definition 4.3. The language $\mathcal{L}^2_{\mathsf{PA}}$ denotes the extension of $\mathcal{L}_{\mathsf{PA}}$ to the language of second-order arithmetic, i.e., it is $\mathcal{L}_{\mathsf{PA}}$ extended by a second sort of variables X, referred to as *set variables* or just *sets*.

Notation 4.4. We use the following notations for certain symbols of $\mathcal{L}_{\mathsf{PA}}$:

- $0_{\mathbb{N}}$ and $\mathbf{0}$ denote the constant $\mathbf{0}^0$ for the number zero,
- $+_{\mathbb{N}}$ denotes the binary function symbol for addition of two natural numbers,
- $<_{\mathbb{N}}$ denotes the binary less-than relation on the natural numbers, and
- $\dot{-}_{\mathbb{N}}$ denotes the modified subtraction function on the natural numbers (i.e, if $m <_{\mathbb{N}} n$ then $m \dot{-}_{\mathbb{N}} n = 0$ holds).

Further, $1_{\mathbb{N}}, 2_{\mathbb{N}}, \dots$ abbreviate $\mathbf{S0}, \mathbf{S}(\mathbf{S0}), \dots$ as usual. If the meaning becomes clear from the context, we may drop the subscript $\mathbb{N}$ and just use $0, 1, 2, \dots, +, <$, and $\dot{-}$ instead. Moreover, $s \leq t$ is used in the obvious way to denote $s < t \vee s = t$.

Definition 4.5. For any $k \in \mathbb{N}$, let $(n_1, \dots, n_k) \mapsto \langle n_1, \dots, n_k \rangle$ be any of the usual primitive recursive injective functions $\mathbb{N}^k \to \mathbb{N}$ mapping finite lists of natural numbers of length k into the natural numbers, and let $\langle\rangle_k$ be the corresponding k-ary function symbol in $\mathcal{L}_{\mathsf{PA}}$. Then for any terms $t_1, \dots, t_k$, we ambiguously write $\langle t_1, \dots, t_k \rangle$ in order to denote $\langle\rangle_k(t_1, \dots, t_k)$.

Moreover, we have the usual primitive recursive functions for projection $(m, n) \mapsto (m)_n$, list construction $(m, n) \mapsto \text{cons}(m, n)$, list concatenation $(m, n) \mapsto m * n$, and for computing the length $n \mapsto \mathsf{lh}(n)$ of a list, which again we use ambiguously to denote the application of its corresponding function symbol in $\mathcal{L}_{\mathsf{PA}}$ to terms. We also make the following standard properties explicit:

- $\langle n_1, \dots, n_k \rangle = 0$ if and only if $k = 0$,
- If $n \neq 0$ holds, then there is exactly one $k \neq 0$ and natural numbers $n_1, \dots, n_k$ such that $n = \langle n_1, \dots, n_k \rangle$ holds,
- $(n)_i < n$ for each $i < \mathsf{lh}(n)$,
- $\mathsf{lh}(\langle n_1, \dots, n_k \rangle) = k$,
- $(\langle n_0, \dots, n_k \rangle)_i = n_i$ for each $0 \leq i \leq k$,
- $\text{cons}(n, \langle n_1, \dots, n_k \rangle) = \langle n, n_1, \dots, n_k \rangle$
- $\langle n_1, \dots, n_k \rangle * \langle m_1, \dots, m_l \rangle = \langle n_1, \dots, n_k, m_1, \dots, m_l \rangle$

Convention 4.6. $\mathcal{L}$ will denote in the following either $\mathcal{L}^2_{\mathsf{PA}}$ or any extension of $\mathcal{L}_{\mathsf{PA}}$ by new relation symbols. We will introduce common notions for such languages $\mathcal{L}$.

Definition 4.7. $\mathcal{L}$ terms s, t, r are defined as usual inductively from function symbols and number variables. Since $\mathcal{L}$ extends $\mathcal{L}_{\mathsf{PA}}$ only by relation symbols or variables of the second sort, all such terms are $\mathcal{L}_{\mathsf{PA}}$ terms. A *constant* is a nullary function symbol. If f is an n-ary function symbol of $\mathcal{L}_{\mathsf{PA}}$ and $\vec{t} = t_1, \dots, t_n$ is a list of terms, then we set

$$f(\vec{t}) := f(t_1, \dots, t_n) := f\vec{t} := ft_1 \dots t_n$$

and this holds analogously for lists introduced by the $\bar{t}^{(n)}$ notation. For closed terms t, we mean by $t^{\mathbb{N}}$ the *numerical value* of t, i.e., the canonical valuation of t in the standard model $\mathbb{N}$.

Definition 4.8. $\mathcal{L}$ *formulas* are defined inductively as usual by use of parentheses and the basic logical symbols and we write ambiguously $A \in \mathcal{L}$ in order to stress that A is an $\mathcal{L}$ formula. For terms s, t, we may sometimes write $s \neq t$ for $\neg(s = t)$. *Atomic* $\mathcal{L}$ *formulas* are equations $s = t$ and all formulas $Rt_1 \dots t_n$ where $R \in \mathcal{L}$ is an n-ary relation symbol and $t_1, \dots, t_n$ are terms.

For the case that $\mathcal{L}$ is $\mathcal{L}^2_{\mathsf{PA}}$, then also Xt is an atomic formula for any set variable X and term t. $\mathcal{L}^2_{\mathsf{PA}}$ formulas further allow for *quantification over set variables* and we call an $\mathcal{L}^2_{\mathsf{PA}}$ formula *arithmetical* if it does not contain such a quantification (but set variables may still occur and we sometimes call set variables that occur free in a formula *set parameters* of this formula).

For n-ary relation symbols (or set variables) R of $\mathcal{L}$, a formula A is *positive in* R if it occurs only positively in the usual sense, i.e., no atomic formula of the form $R(t_1, \dots, t_n)$ occurs negated in the formula which is obtained from A by translating first each subformula of the form $B_1 \to B_2$ to $\neg B_1 \lor B_2$ and where we then move every negation symbol $\neg$ towards atomic formulas, while making use of De Morgan's laws and the law of double negation.[5]

Definition 4.9.

(a) For any language $\mathcal{L}$ that is $\mathcal{L}^2_{\mathsf{PA}}$ or (possibly) extends $\mathcal{L}_{\mathsf{PA}}$ by new relation or function symbols, a standard derivability notion $\vdash$ shall be given that is based on a Hilbert-style deduction system for classical logic with equality axioms (in the first sort). In particular for $\mathcal{L}^2_{\mathsf{PA}}$, we assume besides modus ponens the usual axioms and rules for quantification over set variables.

(b) Then for any $\mathcal{L}$ formula A, we write $\vdash A$ to denote the derivability of A in this logic. Moreover, if T is a theory (i.e., a collection of non-logical axioms) with language $\mathcal{L}_{\mathsf{T}}$, then writing $\mathsf{T} \vdash A$ for any $\mathcal{L}_{\mathsf{T}}$ formula A denotes the derivability of A from the axioms of T and this logic. For any set of formulas Γ, we write $\vdash \Gamma$ and $\mathsf{T} \vdash \Gamma$ in order to denote that $\vdash A$ and $\mathsf{T} \vdash A$ hold, respectively, for each $A \in \Gamma$.

Notation 4.10. For an n-ary relation symbol R with $n \geq 1$ and $\vec{t} = t_1 \dots t_n$, we write $R(\vec{t})$ for $Rt_1 \dots t_n$. and if $n = 1$, we also introduce the following notation:

$$t \in R := Rt \quad \text{and} \quad t \notin R := \neg Rt$$

Then $(\forall x \in R)A$ and $(\exists x \in R)A$ stand for $\forall x(R(x) \to A)$ and $\exists x(R(x) \land A)$, respectively. These conventions shall hold analogously also for set variables X. If $\lhd$ is a binary relation symbol, we use expressions $(\forall x \lhd t)A$ and $(\exists x \lhd t)A$ to abbreviate $\forall x(x \lhd t \to A)$ and $\exists x(x \lhd t \land A)$, respectively.

Definition 4.11. The first-order theory PA is based on the language $\mathcal{L}_{\mathsf{PA}}$ and its non-logical axioms are the usual axioms of Peano arithmetic, while for each relation symbol R_f that stems from a function symbol f of arity $n \geq 1$, we have for $\vec{x} = x_1, \dots, x_n$ the axiom $\forall \vec{x}(R_f \vec{x} \leftrightarrow f\vec{x} = 0)$.

In particular for the formulation of PR as presented in Definition 4.1, the non-logical axioms of PA consist of the universal closure of the following formulas where we suppose $A \in \mathcal{L}_{\mathsf{PA}}$, $(\mathbf{C}fg_1 \dots g_m) \in \mathrm{PR}^n$, and $(\mathbf{R}fg) \in \mathrm{PR}^{n+1}$:

[5] Compare this definition of positive formula with the definition of For^+ in the setting of FIT.

$$
\begin{aligned}
&\mathbf{S}x \neq 0 \\
&\mathbf{S}x = \mathbf{S}y \to x = y \\
&\mathbf{0}^n x_1 \dots x_n = 0 \\
&\mathbf{I}^n_i x_1 \dots x_n = x_i \\
&(\mathbf{C} f g_1 \dots g_m) x_1 \dots x_n = f(g_1 x_1 \dots x_n) \dots (g_m x_1 \dots x_n) \\
&(\mathbf{R} f g) x_1 \dots x_n 0 = f x_1 \dots x_n \\
&(\mathbf{R} f g) x_1 \dots x_n (\mathbf{S}y) = g x_1 \dots x_n y ((\mathbf{R} f g) x_1 \dots x_n y) \\
&R_f x_1 \dots x_n \leftrightarrow f x_1 \dots x_n = 0 \\
&A(0/x) \to \forall x (A \to A(\mathbf{S}x/x)) \to \forall x A \qquad (\textit{complete induction})
\end{aligned}
$$

There is no non-logical axiom for the unary relation symbol U (besides in an instance of complete induction).

Definition 4.12. *(Arithmetical) operator forms* are objects of the form

$$\Lambda X.\mathcal{A}$$

for $\mathcal{L}^2_{\mathsf{PA}}$ class terms of the form $\mathcal{A} = \Lambda x.A$ such that A is an arithmetical formula with X being the *only* set variable that may occur in it (compare also with Section 2) and x is the *only* free number variable that may occur in it.[6] Note that the unary relation symbol U may occur in $\mathfrak{A}$. We use $\mathfrak{A}, \mathfrak{B}, \mathfrak{C}, \mathfrak{D}$ as syntactic variables for operator forms. For each $\mathcal{L}$ class terms $\mathcal{B}$, we set

$$(\Lambda X.\mathcal{A})(\mathcal{B}) := \mathcal{A}(\mathcal{B}/X)$$

while note that the expression $\mathcal{A}(\mathcal{B}/X)$ may yield an $\mathcal{L}$ formula here. Moreover, if R is a unary relation symbol in $\mathcal{L}$ or a set variable, then we write $\mathfrak{A}(R)$ to denote $\mathfrak{A}(\Lambda x.Rx)$. *Positive operator forms* are operator forms $\mathfrak{A} := \Lambda X.\Lambda x.A$ such that X occurs only positively in A.

Notation 4.13. We have the following abbreviations for some formulas and operator forms:

- $\mathrm{Cl}_{\mathfrak{A}}(\mathcal{A}) := \forall x(\mathfrak{A}(\mathcal{A}, x) \to \mathcal{A}(x))$ for each operator form $\mathfrak{A}$ and $\mathcal{L}$ class term $\mathcal{A}$.

and for a binary relation symbols $\lhd$ in $\mathcal{L}_{\mathsf{PA}}$ and any class term $\mathcal{A}$, we also have

- $\mathrm{Acc}_{\lhd} := \Lambda X.\Lambda x.\forall y \lhd x(Xy)$,
- $\mathrm{Prog}_{\lhd}(\mathcal{A}) := \mathrm{Cl}_{\lhd}(\mathcal{A}) := \mathrm{Cl}_{\mathrm{Acc}_{\lhd}}(\mathcal{A})$,
- $\mathrm{TI}_{\lhd} := \Lambda X.\Lambda x.(\mathrm{Prog}_{\lhd}(X) \to \forall y \lhd x(Xy))$, and

Note that we shall usually write $\mathrm{Prog}_{\lhd}$ instead of $\mathrm{Cl}_{\lhd}$. If $\lhd$ is clear from the context, we may just write Acc, Cl, Prog, and TI instead of $\mathrm{Acc}_{\lhd}$, $\mathrm{Cl}_{\lhd}$, $\mathrm{Prog}_{\lhd}$, and $\mathrm{TI}_{\lhd}$, respectively.

[6] Recall that $\mathfrak{A} := \Lambda X.\Lambda x.A$ is intended to define an operator $\Phi_{\mathfrak{A}} \colon \mathcal{P}(\mathbb{N}) \to \mathcal{P}(\mathbb{N})$ where $\mathfrak{A}(X, x)$ corresponds to "$x \in \Phi_{\mathfrak{A}}(X)$" for some interpretation $X \subseteq \mathbb{N}$ and $x \in \mathbb{N}$ of x and X.

4.1. The Accessible Part Theory TID

Definition 4.14.

(a) For each operator form $\mathfrak{A}$, let $P_{\mathfrak{A}}$ denote a new unary relation symbol not in $\mathcal{L}_{\mathsf{PA}}$. Then, $P_{\lhd}$ abbreviates $P_{\mathrm{Acc}_{\lhd}}$ for any binary relation symbol $\lhd$ in $\mathcal{L}_{\mathsf{PA}}$.

(b) The language of TID is defined as

$$\mathcal{L}_{\mathsf{TID}} := \mathcal{L}_{\mathsf{PA}} \cup \{P_{\lhd} : \lhd \text{ is a binary relation symbol in } \mathcal{L}_{\mathsf{PA}}\}$$

Definition 4.15 (Pos_0 and $\mathrm{Pos}_1(a)$)**.** We first set

$$\mathrm{Pos}_0 := \{A \in \mathcal{L}_{\mathsf{TID}} \colon P_{\lhd} \text{ occurs at most positively in } A \text{ for any binary relation symbol } \lhd \text{ in } \mathcal{L}_{\mathsf{PA}}\}$$

and then define $\mathrm{Pos}_1(a)$ for any number variable a as the collection of $\mathcal{L}_{\mathsf{TID}}$ formulas A such that one of the following cases holds:

- $A \in \mathrm{Pos}_0$
- $A = \forall \vec{x}(B_1 \to B_2)$ with
  - $a \notin \mathrm{FV}(B_1)$,
  - $B_1, B_2 \in \mathrm{Pos}_0$, and
  - $\vec{x}$ being a (possibly empty) list of variables.

This is motivated by FT from the setting of FIT.

Remark 4.16. The variable condition $a \notin \mathrm{FV}(B_1)$ in the definition of $\mathrm{Pos}_1(a)$ could have been omitted (as described in [Ran15]). We kept this condition in order to have a more direct relationship between TID and FIT.

Definition 4.17 (Neg_0)**.** Let $\mathrm{Neg}_0 := \{A \in \mathcal{L}_{\mathsf{TID}} \colon \neg A \in \mathrm{Pos}_0\}$.

Notation 4.18. We write $\Lambda a.A \in \mathrm{Pos}_1$ in order to denote $A \in \mathrm{Pos}_1(a)$.

Example 4.19. Let f be some binary function symbol in $\mathcal{L}_{\mathsf{PA}}$ and a a number variable. Then $\mathrm{Pos}_1(a)$ contains the formula $A := \forall y(P_{\lhd} y \to \forall x \lhd a(P_{\lhd} fxy))$ and we have $\Lambda a.A \in \mathrm{Pos}_1$.

Definition 4.20 (TID)**.** TID is the theory that arises from the axioms of Peano arithmetic PA without complete induction by adding the following axioms and axiom schemes

(Ind) $\mathcal{B}(0) \land \forall x(\mathcal{B}(x) \to \mathcal{B}(\mathbf{S}x)) \to \forall x \mathcal{B}(x)$
for $\mathcal{B} \in \mathrm{Pos}_1$

(Cl) $\mathrm{Prog}_{\lhd}(P_{\lhd})$ (i.e., $\forall x(\mathrm{Acc}_{\lhd}(P_{\lhd}, x) \to P_{\lhd} x)$)
for $\lhd$ being a binary relation symbol in $\mathcal{L}_{\mathsf{PA}}$

(TID) $\mathrm{Prog}_{\lhd}(\mathcal{B}) \to \forall x(P_{\lhd} x \to \mathcal{B}(x))$
for $\mathcal{B} \in \mathrm{Pos}_1$ and $\lhd$ being a binary relation symbol in $\mathcal{L}_{\mathsf{PA}}$

where (Cl) is called *closure* and (TID) is called *typed inductive definition.*

Remark 4.21. For any binary relation symbol $\lhd$ in $\mathcal{L}_{\mathsf{PA}}$, we may identify (CI) with a fixed-point principle

$$(\mathsf{FP}) \qquad \forall x(P_{\lhd}x \leftrightarrow \mathrm{Acc}_{\lhd}(P_{\lhd}, x))$$

and therefore we will sometimes use (CI) to ambiguously mean (FP).

Abbreviating $P_{\lhd}$ by P and $\mathrm{Acc}_{\lhd}$ by Acc, we explain how (FP) follows from TID: We get $\forall x(Px \to \mathrm{Acc}(P, x))$ by (TID) with $\mathcal{B} := \Lambda a.\mathrm{Acc}(P, a)$, first because $\mathrm{Acc}(P, a)$ equals $\forall y(y \lhd a \to Py)$ which is in $\mathrm{Pos}_0(a)$, and second because $\forall x(\mathrm{Acc}(\mathcal{B}, x) \to \mathcal{B}(x))$ holds by using (CI) and that Acc is a positive operator form.

Remark 4.22. We can use instead of (Ind) also the following course-of-value variant of complete induction for Pos_1 formulas, i.e., we have

$$\forall x(\forall x_0 <_{\mathbb{N}} x\mathcal{B}(x_0) \to \mathcal{B}(x)) \to \forall x\mathcal{B}(x)$$

as an induction principle for all $\mathcal{B} \in \mathrm{Pos}_1$. In the following, we shall make use of this variant without mentioning it explicitly.

4.2. Embedding TID into FIT

Definition 4.23. For each $f \in \mathrm{PR}^n$ and $n \in \mathbb{N}$, we define an $\mathcal{L}_{\mathsf{FIT}}$ term pr_f recursively on the build-up of f (and where we let $\vec{x} = x_1, \ldots, x_n$):

$\lambda\vec{x}.0$	if $f = \mathbf{0}^n$
s_{N}	if $f = \mathbf{S}$
$\lambda\vec{x}.x_i$	if $f = \mathbf{I}^n_i$
$\lambda\vec{x}.\mathsf{pr}_g(\mathsf{pr}_{h_1}\vec{x})\ldots(\mathsf{pr}_{h_m}\vec{x})$	if $f = (\mathbf{C}gh_1\ldots h_m)$
$\lambda\vec{x}.\mathsf{fix}(t_{g,h}\bar{x}^{(n-1)})x_n$	if $f = (\mathbf{R}gh)$

where

$$t_{g,h} := \begin{cases} \lambda\bar{x}^{(n-1)}h_0x_n. \\ \mathsf{d}_{\mathsf{N}}\mathsf{pr}_g\big(\lambda\bar{z}^{(n-1)}.\mathsf{pr}_h\bar{z}^{(n-1)}(\mathsf{p}_{\mathsf{N}}x_n)(h_0\bar{z}^{(n-1)}(\mathsf{p}_{\mathsf{N}}x_n))\big)0x_n\bar{x}^{(n-1)} \end{cases}$$

and fix is the closed term given in Lemma 3.13.(b).

Theorem 4.24. *For each n-ary function symbol $f \in \mathcal{L}_{\mathsf{PA}}$, we have the following.*

(a) FIT *proves the reformulation of every defining equation of f from Definition 4.11 with respect to pr_f, while interpreting number variables x as individual variables x with $x \in \mathsf{N}$.*

(b) $\mathsf{FIT} \vdash \mathsf{pr}_f \in \mathsf{N}^n \twoheadrightarrow \mathsf{N}$.

Proof. It is straight-forward to verify (a) and (b) by induction on the build-up of $f \in \mathrm{PR}^n$, given the translation from Definition 4.23 and by making use of the induction principle (FT-Ind). $\square$

Definition 4.25. Based on the translation given in Definition 4.23, we define for each $\mathcal{L}_{\mathsf{PA}}$ term t the translation $t^\bullet$ to an $\mathcal{L}_{\mathsf{FIT}}$ term recursively on the build-up of t:

x	if t is a variable x
pr_c	if t is a constant c
$\mathsf{pr}_f t_1^\bullet \dots t_n^\bullet$	if t is $ft_1 \dots t_n$ with $f \in \mathrm{PR}^n$ and $n \geq 1$

The translation on terms is now extended to $\mathcal{L}_{\mathsf{TID}}$ formulas A. We define the $\mathcal{L}_{\mathsf{FIT}}$ formula $A^\bullet$ recursively on the build-up of an $\mathcal{L}_{\mathsf{TID}}$ formula A:

$s^\bullet = t^\bullet$	if A is $s = t$
$\mathsf{pr}_f t_1^\bullet \dots t_n^\bullet = 0$	if A is $R_f t_1 \dots t_n$
$t^\bullet \in \mathsf{U}$	if A is $\mathsf{U}t$
$t^\bullet \in \mathsf{I}_{\mathsf{N},\mathbb{Q}_\lhd}$	if A is $P_\lhd t$ and where $\mathbb{Q}_\lhd := \{\langle x, y\rangle \colon (x \lhd y)^\bullet\}$
$\neg(B^\bullet)$	if A is $\neg B$
$B^\bullet \circ C^\bullet$	if A is $B \circ C$ for $\circ \in \{\wedge, \vee, \rightarrow\}$
$\forall x(x \in \overline{\mathsf{N}} \vee B^\bullet)$	if A is $\forall x B$ (see also Remark 4.26)
$\exists x(x \in \mathsf{N} \wedge B^\bullet)$	if A is $\exists x B$

The expression $\{\langle x, y\rangle \colon (x \lhd y)^\bullet\}$ is a short-hand notation for the expression

$$\{z \colon z = \langle (z)_0, (z)_1\rangle \wedge (x \lhd y)^\bullet((z)_0/x, (z)_1)/y\}$$

i.e., for $\{z \colon z = \langle (z)_0, (z)_1\rangle \wedge R_f(z)_0(z)_1\}$ where f is such that R_f is $\lhd$.

Remark 4.26.

(a) Note that we defined $A^\bullet$ in case of A being $\forall x B$ in such a way that we can show that $A \in \mathrm{For}^+$ implies $A^\bullet \in \mathrm{Pos}_0$ in Lemma 4.27. If we would have defined it to be $\forall x(x \in \mathsf{N} \rightarrow B^\bullet)$, then apparently N would occur negatively in $A^\bullet$.

(b) It can be readily checked that $A^\bullet$ is indeed a $\mathcal{L}_{\mathsf{FIT}}$ formula. Moreover, A and $A^\bullet$ have the same free variables. In particular, note that $(x \lhd y)^\bullet((z)_0/x, (z)_1)/y\}$ contains only z as a free variable.

(c) We will use the expression $\mathbb{Q}_\lhd$ without further mentioning in order to denote the type that we introduced in the definition of $(P_\lhd t)^\bullet$. Recall also that $(x \lhd y)^\bullet$ equals $\mathsf{pr}_f xy = 0$ for some binary function symbol $f \in \mathcal{L}_{\mathsf{PA}}$ because the binary relation symbol $\lhd \in \mathcal{L}_{\mathsf{PA}}$ is of the form R_f for such an f.

Lemma 4.27.

(a) For each $A \in \mathrm{Pos}_0$ there is a formula $A' \in \mathrm{For}^+$ with $\mathrm{FV}(A^\bullet) = \mathrm{FV}(A')$ and such that $\mathsf{FIT} \vdash A^\bullet \leftrightarrow A'$ holds.

(b) For each $A \in \mathrm{Neg}_0$, there is a formula $A' \in \mathrm{For}^+$ with $\mathrm{FV}(A^\bullet) = \mathrm{FV}(A')$ and such that $\mathsf{FIT} \vdash A^\bullet \leftrightarrow \neg A'$ holds.

Proof. By simultaneous induction on the build-up of A. □

Definition 4.28. For every $A \in \mathcal{L}_{\mathsf{TID}}$, we define

$$A^\bullet_{\mathsf{N}} := \begin{cases} A^\bullet & \text{if } \mathrm{FV}(A) = \emptyset \\ x_1 \in \mathsf{N} \to \ldots \to x_n \in \mathsf{N} \to A^\bullet & \text{if } \mathrm{FV}(A) = \{x_1, \ldots, x_n\} \\ & \quad \text{for some } n \neq 0 \end{cases}$$

Lemma 4.29. *For each $B \in \mathrm{Pos}_1(a)$, there is an $\mathcal{L}_{\mathsf{FIT}}$-term t and a function type $\mathbb{F} \in \mathrm{FT}$ such that*

$$\mathsf{FIT} \vdash \forall x(tx \in \mathbb{F} \leftrightarrow B^\bullet(x/a))$$

holds.

Proof. We distinguish the following cases on $B \in \mathrm{Pos}_1(a)$:
1. If $B \in \mathrm{Pos}_0$, then Lemma 4.27 provides some $B' \in \mathrm{For}^+$ such that $\mathsf{FIT} \vdash B^\bullet \leftrightarrow B'$ holds, so for $\mathbb{F} := \{a \colon B'\}$ we have $\mathbb{F} \in \mathrm{FT}$. Moreover, with $t := \lambda x.x$, we get the claim.
2. If B is of the form $\forall \vec{y}(B_1 \to B_2)$ with $a \notin \mathrm{FV}(B_1)$, $\vec{y} = y_1, \ldots, y_n$, and $B_1, B_2 \in \mathrm{Pos}_0$, we first get $B_1', B_2' \in \mathrm{For}^+$ from Lemma 4.27 such that

$$B_i^\bullet \leftrightarrow B_i' \;\;\&\;\; \mathrm{FV}(B_i^\bullet) = \mathrm{FV}(B_i') \qquad (i = 1, 2)$$

and then we set

$$\begin{aligned} \mathbb{Q}_1 &:= \{z \colon z = \langle (z)_0, \ldots, (z)_{n-1}\rangle \wedge B_1'((z)_0/y_1, \ldots, (z)_{n-1}/y_n)\} \\ \mathbb{Q}_2 &:= \{z \colon z = \langle (z)_0, \ldots, (z)_n\rangle \wedge B_2'((z)_n/a, (z)_0/y_1, \ldots, (z)_{n-1}/y_n)\} \\ \mathbb{F} &:= \mathbb{Q}_1 \twoheadrightarrow \mathbb{Q}_2 \\ t &:= \lambda x, z.\langle x, (z)_0, \ldots, (z)_{n-1}\rangle \end{aligned}$$

Obviously $\mathbb{F} \in \mathrm{FT}$ holds and then similar as in [Fef92, 6.3], we have over FIT and for any x

$$\begin{aligned} tx \in \mathbb{F} &\leftrightarrow \forall z(z \in \mathbb{Q}_1 \to txz \in \mathbb{Q}_2) \\ &\leftrightarrow \forall \vec{y}(B_1^\bullet \to tx\langle y_1, \ldots, y_n\rangle \in \mathbb{Q}_2) \\ &\leftrightarrow \forall \vec{y}(B_1^\bullet \to \langle x, y_1, \ldots, y_n\rangle \in \mathbb{Q}_2) \\ &\leftrightarrow \forall \vec{y}(B_1^\bullet \to B_2^\bullet(x/a)) \\ &\leftrightarrow B^\bullet(x/a) \end{aligned}$$

which gives us the claim. Note that $n = 0$ is possible, so $\forall \vec{y}(B_1^\bullet \to tx\langle y_1, \ldots, y_n\rangle \in \mathbb{Q}_2)$ denotes then $B_1^\bullet \to (tx\langle\rangle \in \mathbb{Q}_2)$. □

Theorem 4.30. FIT *proves every translation $A^\bullet$ of an instance A of axioms* (Ind), (Cl), *and* (TID) *from* TID_1. *More precisely, if A is an instance of* (Ind), (Cl), *or* (TID), *then we have* $\mathsf{FIT} \vdash A^\bullet$.

Proof. Let A be an instance of (Ind), (Cl), or (TID). We have to show $\mathsf{FIT} \vdash A^\bullet$.
1. For (Cl): If $A = \mathrm{Prog}_\lhd(P_\lhd)$ holds for some $\lhd$, then we have that $A^\bullet$ is logically equivalent over FIT to $\mathrm{Cl}_{\mathsf{N},\mathbb{Q}_\lhd}(\Lambda z.z \in \mathsf{I}_{\mathsf{N},\mathbb{Q}_\lhd})$, and this is an instance of (FT-Cl). More precisely, we have over FIT:

$$
\begin{aligned}
&\big(\mathrm{Prog}_{\lhd}(P_{\lhd})\big)^{\bullet}\\
&\leftrightarrow \big(\forall x(\mathrm{Acc}_{\lhd}(P_{\lhd},x) \to P_{\lhd}x)\big)^{\bullet}\\
&\leftrightarrow \forall x(x \in \overline{\mathsf{N}} \vee \big((\mathrm{Acc}_{\lhd}(P_{\lhd},x))^{\bullet} \to x \in \mathsf{I}_{\mathsf{N},\mathbb{Q}_{\lhd}}\big))\\
&\leftrightarrow \forall x(x \in \overline{\mathsf{N}} \vee \big(\forall y(y \in \overline{\mathsf{N}} \vee \big((y \lhd x)^{\bullet} \to y \in \mathsf{I}_{\mathsf{N},\mathbb{Q}_{\lhd}}\big)\big) \to x \in \mathsf{I}_{\mathsf{N},\mathbb{Q}_{\lhd}}\big))\\
&\leftrightarrow \forall x(x \in \overline{\mathsf{N}} \vee \big(\forall y(y \in \overline{\mathsf{N}} \vee \big(\langle y,x\rangle \in \mathbb{Q}_{\lhd} \to y \in \mathsf{I}_{\mathsf{N},\mathbb{Q}_{\lhd}}\big)\big) \to x \in \mathsf{I}_{\mathsf{N},\mathbb{Q}_{\lhd}}\big))\\
&\leftrightarrow (\forall x \in \mathsf{N})((\forall y \in \mathsf{N})(\langle y,x\rangle \in \mathbb{Q}_{\lhd} \to y \in \mathsf{I}_{\mathsf{N},\mathbb{Q}_{\lhd}}) \to x \in \mathsf{I}_{\mathsf{N},\mathbb{Q}_{\lhd}})\\
&\leftrightarrow \mathrm{Cl}_{\mathsf{N},\mathbb{Q}_{\lhd}}(\Lambda z.z \in \mathsf{I}_{\mathsf{N},\mathbb{Q}_{\lhd}})
\end{aligned}
$$

2. For (Ind) and (TID): Let $B \in \mathrm{Pos}_1(a)$ be arbitrary. By Lemma 4.29 some $\mathcal{L}_{\mathsf{FIT}}$-term t and function type $\mathbb{F} \in \mathrm{FT}$ exist such that we have

$$
\mathsf{FIT} \vdash \forall x(tx \in \mathbb{F} \leftrightarrow B^{\bullet}(x/a)) \tag{1}
$$

2.1. If $A = \mathcal{B}(0) \wedge \forall x(\mathcal{B}(x) \to \mathcal{B}(\mathbf{S}x)) \to \forall x\mathcal{B}(x)$ holds for $\mathcal{B} = \Lambda a.B$: We note that for $B_1 := B(a/\mathbf{S}a)$ one can prove (by induction on the build-up of B) that $B_1^{\bullet}$ is $B^{\bullet}(a/\mathsf{s_N}a)$. So, with $\mathcal{B}(\mathbf{S}x)^{\bullet}$ being $(B(a/\mathbf{S}x))^{\bullet}$ this becomes $(B_1(x))^{\bullet}$, i.e., we get $B_1^{\bullet}(a/x)$ and hence $(B^{\bullet}(a/\mathsf{s_N}a))(a/x)$. So, we obtain that $\mathcal{B}(\mathbf{S}x)^{\bullet}$ is $B^{\bullet}(a/\mathsf{s_N}x)$, while note that for any $B' \in \mathcal{L}_{\mathsf{TID}}$, we have that B' and $B'^{\bullet}$ share the same first-order variables. For proving $A^{\bullet}$, we can therefore assume that

$$
B^{\bullet}(0/a) \tag{2}
$$

$$
\forall x(x \in \overline{\mathsf{N}} \vee (B^{\bullet}(x/a) \to B^{\bullet}(\mathsf{s_N}x/a))) \tag{3}
$$

holds, and we have to show $\forall x(x \in \overline{\mathsf{N}} \vee B^{\bullet}(x/a))$, while this is equivalent to $t \in \mathsf{N} \twoheadrightarrow \mathbb{F}$ due to (1). Now we can directly apply (FT-Ind) because (2) is equivalent to $t0 \in \mathbb{F}$ and (3) is equivalent to $(\forall x \in \mathsf{N})(tx \in \mathbb{F} \to t(\mathsf{s_N}) \in \mathbb{F})$.

2.2. If $A = \mathrm{Prog}_{\lhd}(\mathcal{B}) \to \forall x(P_{\lhd}x \to \mathcal{B}(x))$ holds for $\mathcal{B} = \Lambda a.B$: With

$$
\big(\mathrm{Prog}_{\lhd}(\mathcal{B})\big)^{\bullet} = (\forall x \in \mathsf{N})(\forall y \in \mathsf{N})\big(\langle y,x\rangle \in \mathbb{Q}_{\lhd} \to \big(\mathcal{B}(y)\big)^{\bullet} \to \big(\mathcal{B}(x)\big)^{\bullet}\big) \tag{4}
$$

we get that FIT proves the following:

$$
\left.
\begin{aligned}
&\big(\mathrm{Prog}_{\lhd}(\mathcal{B})\big)^{\bullet}\\
&\leftrightarrow (\forall x \in \mathsf{N})(\forall y \in \mathsf{N})(\langle y,x\rangle \in \mathbb{Q}_{\lhd} \to ty \in \mathbb{F} \to tx \in \mathbb{F})\\
&\leftrightarrow \mathrm{Cl}_{\mathsf{N},\mathbb{Q}_{\lhd}}(\Lambda z.tz \in \mathbb{F})
\end{aligned}
\right\} \tag{5}
$$

This accumulates in the provability of $A^{\bullet}$. Namely, assume $\big(\mathrm{Prog}_{\lhd}(\mathcal{B})\big)^{\bullet}$ and get $t \in (\mathsf{I}_{\mathsf{N},\mathbb{Q}_{\lhd}} \twoheadrightarrow \mathbb{F})$ from (5) and (FT-ID), hence (1) yields

$$
\begin{aligned}
\forall x(x \in \mathsf{I}_{\mathsf{N},\mathbb{Q}_{\lhd}} \to tx \in \mathbb{F}) &\leftrightarrow \forall x(x \in \mathsf{I}_{\mathsf{N},\mathbb{Q}_{\lhd}} \to B^{\bullet}(x/a))\\
&\to (\forall x \in \mathsf{N})(x \in \mathsf{I}_{\mathsf{N},\mathbb{Q}_{\lhd}} \to B^{\bullet}(x/a))
\end{aligned}
$$

Now, we are done because $\big(\forall x(P_{\lhd}x \to \mathcal{B}(x))\big)^{\bullet}$ is $(\forall x \in \mathsf{N})(x \in \mathsf{I}_{\mathsf{N},\mathbb{Q}_{\lhd}} \to B^{\bullet}(x/a))$. □

Corollary 4.31 (Embedding TID into FIT)**.** *Let $A \in \mathcal{L}_{\mathsf{TID}}$ with $\mathrm{FV}(A) = \{x_1, \ldots, x_n\}$. Then we have*

$$
\mathsf{TID} \vdash A \implies \mathsf{FIT} \vdash A^{\bullet}_{\mathsf{N}}
$$

Proof. The claim follows essentially from Theorems 4.24 and 4.30. In particular, we remark that for FIT, the propositional logical rules and axioms and the quantificational logic for individual variables correspond (under the translation of Definition 4.25) to first-order predicate logic in the setting of TID. □

5. Ordinals

In order to determine the lower bound for the proof-theoretic ordinal of both FIT and TID, we shall carry out wellordering proofs within TID in Section 6. We want to keep the focus of this article on the theories FIT and TID, so we shall not introduce in the main part of the article the underlying ordinal notation system $(\mathrm{OT}, \prec)$ and all its properties that are needed to carry out the wellordering proofs. Instead, we refer to the appendix of this article where the definition of $(\mathrm{OT}, \prec)$ and its properties is given but without proofs. For the proofs, we refer to [Ran15].

The ordinal notation system $(\mathrm{OT}, \prec)$ from Subsection A.1 in the appendix is based on a framework for representing ordinals that we shall introduce in this section. We shift to the literature most of the preparatory work that is needed to formulate the ordinals that are involved here and try to explain only as much as to make this article sufficiently self-contained. In particular, we use [Sch54] and the treatment of *Klammersymbols* (that extends the concept of finitary Veblen functions to the transfinite) as the main source for the finitary Veblen functions because it is the most elaborated source for representation and recursion properties.

5.1. Ordinal Notations

In this subsection, we work in the broad set-theoretic framework of Zermelo–Fraenkel set theory ZFC with the axiom of choice, having the class On of ordinals at hand and where $\alpha, \beta, \gamma, \delta, \dots$ (small Greek letters in general) are used to denote elements of On. The class of limit ordinals is denoted by Lim, while ω denotes the first limit ordinal. Moreover, we write 0 for $\emptyset$, $\alpha <_{\mathrm{On}} \beta$ (or just $\alpha < \beta$) for $\alpha \in \beta$, and $\alpha \leq_{\mathrm{On}} \beta$ (or just $\alpha \leq \beta$) for $\alpha \subseteq \beta$. For $\alpha > 0$, we let Ω_α denote $\aleph_\alpha$, i.e., $\{\Omega_\alpha \colon \alpha \in \mathrm{On}\}$ is the class of all uncountable initial ordinals, and we write Ω for Ω_1 and Ω_0 for 0. Over ZFC, we have that $\Omega_{\alpha+1}$ is regular. A *normal function* is a (with respect to $<$) strictly increasing continuous function $f \colon \mathrm{On} \to \mathrm{On}$. We presuppose a knowledge about this broad set-theoretic framework and shall use commonly used notions and well-known properties of those tacitly, e.g.,

- the notion of *club* classes C with $C \subseteq \mathrm{On}$ and its correspondence to normal functions (i.e., each club class C induces a normal function enum_C that enumerates the elements of C in increasing order),
- the existence of the *derivative* $\mathrm{fix}(f) := \{\alpha \in \mathrm{On} \colon f(\alpha) = \alpha\}$ of a normal function f, being a club class itself,
- *basic ordinal arithmetic* for $\alpha, \beta \in \mathrm{On}$ with (ordinal) addition $\alpha +_{\mathrm{On}} \beta$ (or just $\alpha + \beta$), (ordinal) multiplication $\alpha \cdot_{\mathrm{On}} \beta$ (or just $\alpha \cdot \beta$ or $\alpha\beta$), and (ordinal) exponentiation $\exp_{\mathrm{On}}(\alpha, \beta)$ (or just α^β),
- the usual representation of natural numbers as *von Neumann ordinals* $(0)_{\mathrm{On}} := \emptyset$ and $(n+1)_{\mathrm{On}} := \{(n)_{\mathrm{On}}\} \cup (n)_{\mathrm{On}}$ for each $n \in \mathbb{N}$ within On, while we shall from now on identify $(n)_{\mathrm{On}}$ with n for each $n \in \mathbb{N}$.

We refer also to [Buc16] for more details on the relationship between different approaches to ordinal notations. It shall be clear from the context whether $<$ means $<_{\mathbb{N}}$ or $<_{\mathrm{On}}$ (and similar for the other mentioned expressions).

Definition 5.1. Let $\mathbf{P} := \{\omega^\eta \colon \eta \in \mathrm{On}\}$. We call the elements of $\mathbf{P}$ *additive principal numbers.*

Remark. For $\alpha \in \mathbf{P}$ and $\beta, \gamma < \alpha$, we have $\beta + \gamma < \alpha$ and $\beta + \alpha = \alpha$.

The finitary Veblen functions

Definition 5.2. The *$n+1$-ary Veblen function* $\varphi_{n+1} : \mathrm{On}^{n+1} \to \mathrm{On}$ is obtained for each $n \in \mathbb{N}$ from the ω-exponential function and the binary Veblen function φ_2 by generalizing its definition principle, i.e., we let

$$\varphi_1(\gamma) := \omega^\gamma$$

for each $\gamma \in \mathrm{On}$ and define φ_{n+2} for $n \geq 0$ as follows:

- $\varphi_{n+2}(0, \bar{\alpha}^{(n)}, \gamma) := \varphi_{n+1}(\bar{\alpha}^{(n)}, \gamma)$.
- If $\alpha_1, \alpha_k > 0$ holds for some $1 \leq k \leq n+1$ with $\alpha_{k+1} = \cdots = \alpha_{n+1} = 0$, then $\varphi_{n+2}(\bar{\alpha}^{(n+1)}, \gamma)$ denotes the γ-th common fixed-point of the functions

  $$\xi \mapsto \varphi_{n+2}(\bar{\alpha}^{(k-1)}, \beta, \xi, \bar{0}^{(n-k+1)})$$

  that are defined on On and for each $\beta < \alpha_k$.

In particular, we have for the binary Veblen function that $\varphi_2(\alpha, \gamma)$ for $\alpha \in \mathrm{On} \setminus \{0\}$ is the γ-th common fixed-point of the functions $\xi \mapsto \varphi_2(\beta, \xi)$ on On and that are given for each $\beta \in \mathrm{On}$ with $\beta < \alpha$.

Definition 5.3. The *small Veblen ordinal* is the least ordinal $\alpha > 0$ not expressible from ordinals smaller than α and by means of ordinal addition and the finitary Veblen functions.

Notation 5.4. We often just write φ instead of φ_{n+1} if the arity $n+1$ is clear from the context.

Remark. We have that $\varphi(0, 1, 0, 0)$ and $\varphi(1, 0, 0)$ denote the Feferman-Schütte ordinal Γ_0 and $\varphi(1, 0)$ denotes the ordinal ε_0.

Lemma 5.5. *Let $k, l \in \mathbb{N}$ and $\alpha_1, \ldots, \alpha_k \in \mathrm{On}$ be given with $\alpha_1 \neq 0$ and $\alpha_k \neq 0$. Then*

$$\big(\beta < \alpha_k \ \ \& \ \ \xi = \varphi(\bar{\alpha}^{(k)}, \bar{0}^{(l+1)})\big) \implies \varphi(\bar{\alpha}^{(k-1)}, \beta, \bar{0}^{(i)}, \xi, \bar{0}^{(j)}) = \xi$$

holds for every β, ξ and $i, j \in \mathbb{N}$ with $i + j = l$.

Proof. This follows easily from Definition 5.2. ∎

Klammersymbols

Definition 5.6. Following [Sch54], we introduce Schütte's Klammersymbols[7] which are a generalization of the finitary Veblen functions to the transfinite by allowing arguments to be indexed by ordinals. A *Klammersymbol* K is an expression of the form

$$\binom{\alpha_1 \ \cdots \ \alpha_{n+1}}{\beta_1 \ \cdots \ \beta_{n+1}}$$

for $\alpha_1, \dots, \alpha_{n+1}, \beta_1, \dots, \beta_{n+1} \in \mathrm{On}$ and with the condition

$$0 \leq \beta_1 < \beta_2 < \cdots < \beta_{n+1} \tag{6}$$

We use K as a syntactic variable to denote a Klammersymbol. Two Klammersymbols K_1 and K_2 are defined to be *equal*, denoted $K_1 = K_2$, in case K_1 and K_2 can be transformed into the same Klammersymbol by adding or dropping of columns of the form $\substack{0 \\ \beta}$. We write $K_1 \neq K_1$ in case that K_1 and K_2 are not equal.

Given a normal function $f\colon \mathrm{On} \to \mathrm{On}$ such that $f(0) > 0$, the *value* fK *of a Klammersymbol* K *(under* f*)* is defined as follows:

1. If K is $\binom{\gamma}{0}$, then fK is $f(\gamma)$.
2. If K is $\binom{\gamma \ \alpha_1 \ \cdots \ \alpha_{n+1}}{0 \ \beta_1 \ \cdots \ \beta_{n+1}}$ and $\alpha_i = 0$ for some $i \in \{1, \dots, n+1\}$, then fK is fK' where K' is obtained from K by deleting the column $\substack{0 \\ \beta_i}$.
3. If K is $\binom{\gamma \ \alpha_1 \ \cdots \ \alpha_{n+1}}{0 \ \beta_1 \ \cdots \ \beta_{n+1}}$ and $\alpha_i \neq 0$ for all $i \in \{1, \dots, n+1\}$, then fK is the γ-th common solution η for the following equations and for all $\alpha' < \alpha_2$ and $\beta' < \beta_2$:

$$f\binom{\eta \ \alpha' \ \alpha_3 \ \cdots \ \alpha_{n+1}}{\beta' \ \beta_2 \ \beta_3 \ \cdots \ \beta_{n+1}} = \eta$$

Definition 5.7. The *large Veblen ordinal* is the least ordinal $\alpha > 0$ not expressible from ordinals smaller than α and by means of ordinal addition and the use of the value of Klammersymbols under the ω-exponential function φ_1.[8]

Remark 5.8. For all Klammersymbols K_1 and K_2, there are ordinals $\alpha_1, \dots, \alpha_{n+1}, \beta_1, \dots, \beta_{n+1}$ and ordinals $\gamma_1, \dots, \gamma_{n+1}$ with $\gamma_1 < \cdots < \gamma_{n+1}$ such that $K_1 = \binom{\alpha_1 \ \cdots \ \alpha_{n+1}}{\gamma_1 \ \cdots \ \gamma_{n+1}}$ and $K_2 = \binom{\beta_1 \ \cdots \ \beta_{n+1}}{\gamma_1 \ \cdots \ \gamma_{n+1}}$, simply by adding or removing of columns of the form $\substack{0 \\ \gamma_i}$ where necessary.

Definition 5.9. A *lexicographic order* $<$ on Klammersymbols is defined for Klammersymbols K_1 and K_2 with $K_1 \neq K_2$ as follows:

1. If $K_1 = \binom{\alpha_1 \ \cdots \ \alpha_{n+1}}{\gamma_1 \ \cdots \ \gamma_{n+1}}$ and $K_2 = \binom{\beta_1 \ \cdots \ \beta_{n+1}}{\gamma_1 \ \cdots \ \gamma_{n+1}}$ and $i \in \{1, \dots, n+1\}$ is the *largest* index with $\alpha_i \neq \beta_i$, then we have $K_1 < K_2$ in case of $\alpha_i < \beta_i$ and $K_2 < K_1$ otherwise.
2. If $K_1 = K_1'$, $K_2 = K_2'$, and $K_1 < K_2$, then also $K_1' < K_2'$.

Proposition 5.10. *Let* $f\colon \mathrm{On} \to \mathrm{On}, \xi \mapsto \omega^\xi$*. Then* $\xi \mapsto f\binom{1}{\xi}$ *is a normal function. In particular, we have* $f\binom{1}{\lambda} = \sup_{\xi<\lambda} f\binom{1}{\xi}$ *for each* $\lambda \in \mathrm{Lim}$.

Proof. See [Sch54, (4.1)–(4.3)]. □

[7] Klammer is the German word for bracket, so Klammersymbols can be read as "bracket symbols".

[8] In Section 8, the conclusion of this article, we shall mention a theory TID_1^+ which has the large Veblen ordinal as its proof-theoretic ordinal.

Proposition 5.11. *Let $f\colon \mathrm{On} \to \mathrm{On}, \xi \mapsto \omega^\xi$ and $K := \left(\begin{smallmatrix}\alpha_1 & \dots & \alpha_{n+1} \\ \gamma_1 & \dots & \gamma_{n+1}\end{smallmatrix}\right)$ be a Klammersymbol. For each Klammersymbol K' with $K < K'$, the following holds:*

(a) $fK = fK'$ holds if and only if $k \in \{1, \dots, n+1\}$ exists such that $\alpha_k = fK'$ and the following holds:

- *$\alpha_i = 0$ for each i with $1 \leq i < k$, and*
- *$\alpha_i < fK'$ for each i with $k < i \leq n+1$.*

(b) $fK < fK'$ holds if $\alpha_i < fK'$ holds for all $i \in \{1, \dots, n+1\}$.

(c) $fK' < fK$ holds if

- *either $fK' < \alpha_k$ holds for some $k \in \{1, \dots, n+1\}$, or*
- *$n \geq 1$ and $j, k \in \{1, \dots, n+1\}$ exist such that $j < k$, $\alpha_j \neq 0$, and $\alpha_k = fK'$.*

Proof. See (7.1)–(7.4) in [Sch54]. Note that the negation of the condition given in (a) yields the conditions stated in (b) and (c). For this, note in particular that $\alpha_k = fK'$ implies $\alpha_k \neq 0$ and hence if $\alpha_i = 0$ holds for each i with $1 \leq i < k$ and the condition of (a) does not hold, then $k < n+1$ holds and i exists with $\alpha_i \geq fK'$ and $k < i \leq n+1$, leading to one of the conditions in (c). □

Lemma 5.12. *Let $f\colon \mathrm{On} \to \mathrm{On}, \xi \mapsto \omega^\xi$. Then we have*

$$\varphi(\alpha_1, \dots, \alpha_{n+1}) = f\left(\begin{smallmatrix}\alpha_{n+1} & \alpha_n & \dots & \alpha_1 \\ 0 & 1 & \dots & n\end{smallmatrix}\right)$$

where we denoted with $0, 1, \dots, n$ in the Klammersymbol's second row ambiguously the corresponding finite ordinals.

Proof. If $n = 0$ or $\alpha_1 = \dots = \alpha_{n+1} = 0$ holds, then the claim is clear. Otherwise, assume $n \neq 0$ and without loss of generality that $\alpha_1 \neq 0$ holds. Further, let $k \in \{1, \dots, n\}$ with $\alpha_k \neq 0$ and $\alpha_{k+1} = \dots = \alpha_n = 0$. The claim now follows by transfinite induction on α_k since $\varphi(\bar{\alpha}^{(n+1)})$ is the α_{n+1}-th common fixed point of the functions

$$\xi \mapsto \varphi(\bar{\alpha}^{(k-1)}, \beta, \xi, \bar{0}^{(n-k)})$$

given for each $\beta < \alpha_k$. Now, we get

$$\begin{aligned}\varphi(\bar{\alpha}^{(k-1)}, \beta, \xi, \bar{0}^{(n-k)}) &= f\left(\begin{smallmatrix}0 & \dots & 0 & \xi & \beta & \alpha_{k-1} & \dots & \alpha_1 \\ 0 & \dots & n-(k+1) & n-k & n-(k-1) & n-(k-2) & \dots & n\end{smallmatrix}\right) \\ &= f\left(\begin{smallmatrix}\xi & \beta & \alpha_{k-1} & \dots & \alpha_1 \\ n-k & n-(k-1) & n-(k-2) & \dots & n\end{smallmatrix}\right)\end{aligned}$$

from the induction hypothesis and for each $\xi \in \mathrm{On}$. Hence the claim follows from Definition 5.6 and Lemma 5.5. □

Corollary 5.13. *Let $n \geq 1$ and ordinals $\alpha_1, \dots, \alpha_n$ be given, then we have the following:*

(a) $\alpha_i \leq \varphi(\bar{\alpha}^{(n)})$ for all $i \in \{1, \dots, n\}$.

(b) If $\alpha_k \neq 0$ for some $k \in \{1, \dots, n\}$, then $\alpha_i < \varphi(\bar{\alpha}^{(n)})$ holds for all $i \in \{1, \dots, k-1\}$.

Proof. This follows from (3.3) and (6.1) in [Sch54] by making use of Lemma 5.12. □

Corollary 5.14. *Let $n \geq 1$ and ordinals $\alpha_1, \ldots, \alpha_n, \beta_1, \ldots, \beta_n$ be given. Then $\varphi(\bar{\alpha}^{(n)}) < \varphi(\bar{\beta}^{(n)})$ holds if and only if some $r \in \{1, \ldots, n\}$ exists such that $\alpha_r \neq \beta_r$ holds with $\alpha_i = \beta_i$ for all $i \in \{1, \ldots, r-1\}$, and such that one of the following holds:*

1. *$\alpha_r < \beta_r$ and $\alpha_i < \varphi(\bar{\beta}^{(n)})$ for all $i \in \{r+1, \ldots, n\}$, or*
2. *$\beta_r < \alpha_r$ and*
   - *either $\varphi(\bar{\alpha}^{(n)}) < \beta_k$ holds for some $k \in \{1, \ldots, n\}$, or*
   - *$\varphi(\bar{\alpha}^{(n)}) = \beta_k$ and $\beta_i \neq 0$ for some $i, k \in \{1, \ldots, n\}$ with $k < i$.*

Proof. This follows immediately from Proposition 5.11 and Lemma 5.12. For the first case $\alpha_r < \beta_r$, note that $\alpha_i < \varphi(\bar{\beta}^{(n)})$ holds anyway for $i \in \{1, \ldots, r\}$ by Corollary 5.13: On the one hand, we have $\beta_r \leq \varphi(\bar{\beta}^{(n)})$ and so $\alpha_r < \varphi(\bar{\beta}^{(n)})$, and on the other hand, $\alpha_r < \beta_r$ also implies $\beta_r \neq 0$ which by Corollary 5.13.(b) gives $\alpha_i = \beta_i < \varphi(\bar{\beta}^{(n)})$ for $i \in \{1, \ldots, r-1\}$. □

Definition 5.15. Let $n \geq 1$.

(a) We say that $\alpha_1, \ldots, \alpha_n$ are in *normal form (w.r.t. φ_n)* in case that $\alpha_i < \varphi(\bar{\alpha}^{(n)})$ holds for each $1 \leq i \leq n$, and we denote this by $\mathrm{NF}^{\varphi}_n(\bar{\alpha}^{(n)})$.

(b) We write $\beta =_{\mathrm{NF}} \varphi(\bar{\alpha}^{(n)})$ in order to denote $\beta = \varphi(\bar{\alpha}^{(n)})$ and $\mathrm{NF}^{\varphi}_n(\bar{\alpha}^{(n)})$.

Remark 5.16. The following lemma describes an important recursion property that is needed for the definition of the ordinal notation system $(\mathrm{OT}, \prec)$ from the appendix. It essentially explains how to obtain an expression $\varphi_n(\bar{\beta}^{(n)})$ such that $\beta_1, \ldots, \beta_n$ are in normal form w.r.t. φ_n, i.e., such that $\beta_i \neq \varphi_n(\bar{\beta}^{(n)})$ holds for each $1 \leq i \leq n$.

Lemma 5.17. *Let $n \geq 1$, $k \in \{1, \ldots, n\}$, and ordinals $\beta_1, \ldots, \beta_n$, $\alpha_1, \ldots, \alpha_n$ be given with $\beta_k =_{\mathrm{NF}} \varphi(\bar{\alpha}^{(n)})$ and $\beta_{k+1} = \cdots = \beta_n = 0$. Then $\mathrm{NF}^{\varphi}_n(\bar{\beta}^{(n)})$ holds if and only if $\alpha_r \neq \beta_r$ holds for some $r \in \{1, \ldots, k\}$ with $\alpha_i = \beta_i$ for all $i \in \{1, \ldots, r-1\}$ and one of the following holds:*

1. *$\alpha_r < \beta_r$, or*
2. *$\beta_r < \alpha_r$ and for some $i \in \{1, \ldots, n\}$, we have $\beta_k \leq \beta_i$.*
   (In particular, it suffices here to have $i \in \{r+1, \ldots, k-1\}$.)

Proof. Note that we have $\beta_k \neq 0$ by our assumption $\beta_k =_{\mathrm{NF}} \varphi(\bar{\alpha}^{(n)})$, and hence by Corollary 5.13 and $\mathrm{NF}^{\varphi}_n(\bar{a}^{(n)})$, we have

$$\alpha_i < \beta_k \leq \varphi(\bar{\beta}^{(n)}) \tag{7}$$

for all $i \in \{1, \ldots, n\}$ and also $\beta_i < \varphi(\bar{\beta}^{(n)})$ for all $i \in \{1, \ldots, k-1\}$. Furthermore, there has to be some $r \in \{1, \ldots, k\}$ such that $\alpha_r \neq \beta_r$ and $\alpha_i = \beta_i$ holds for all $i \in \{1, \ldots, r-1\}$ since we have $\alpha_k < \beta_k$. Recalling the assumption $\beta_{k+1} = \cdots = \beta_n = 0$, we thus have that $\mathrm{NF}^{\varphi}_n(\bar{\beta}^{(n)})$ holds if and only if $\beta_k < \varphi(\bar{\beta}^{(n)})$ holds, i.e., $\varphi(\bar{\alpha}^{(n)}) < \varphi(\bar{\beta}^{(n)})$. By Corollary 5.14 this is equivalent to the following two situations:
<u>1.</u> If $\alpha_r < \beta_r$: We need $\alpha_i < \varphi(\bar{\beta}^{(n)})$ for each $i \in \{r+1, \ldots, n\}$. But this holds anyway as we have noted in (7).

2. If $\alpha_r > \beta_r$: We need some $i \in \{1, \dots, k\}$ such that either $\beta_i > \varphi(\bar{\alpha}^{(n)}) = \beta_k$ holds, or otherwise $\beta_i = \varphi(\bar{\alpha}^{(n)}) = \beta_k$ holds and there is some $i < j \leq k$ such that $\beta_j \neq 0$ holds. In both cases, $i = k$ is trivial, moreover recall that $\beta_k \neq 0$ holds. Hence, $\beta_k \leq \beta_i$ for some $i \in \{1, \dots, n\}$ suffices in this situation. Actually, $i \in \{r+1, \dots, k-1\}$ is enough since otherwise $\beta_k \leq \beta_i$ can never hold: We have that $\beta_i \leq \alpha_r < \varphi(\bar{\alpha}^{(n)}) = \beta_k$ holds for all $1 \leq i \leq r$ and that $\beta_i = 0$ holds for all $k \leq i \leq n$. □

Remark 5.18. In order to simplify the formulation and proof of Lemma 5.17, we took the lists of ordinals $\alpha_1, \dots, \alpha_n$ and $\beta_1, \dots, \beta_n$ to have the same length $n \geq 2$. Clearly, the lemma holds analogously for lists of ordinals with different length ≥ 2 (just add ordinals of the form 0 to the front of the shorter list to make them the same length).

The ϑ-function

We refer to [RW93] and [Buc16] for more details on the definition and properties of the function $\vartheta\colon \mathrm{On} \to \mathrm{On}$. We state here only the notation $\Omega(n, x)$ that is defined for each $x \in \mathrm{On}$ and each $n \in \mathbb{N}$ by

$$\Omega(0, x) := \Omega \cdot x \qquad \Omega(1, x) := \Omega^x \qquad \Omega(n+1, x) := \Omega^{\Omega(n,x)}$$

and cite the following properties in order to relate the ϑ-function to the setting of Klammersymbols when we present results from the literature in Section 7. We remark that by using the Buchholz ψ-functions from [BS88] or the Feferman-Aczel θ-functions from [Bri75], we also have the following correspondence $\vartheta\Omega^\omega = \psi_0\Omega^{\Omega^\omega} = \theta\Omega^\omega 0$. See also the last paragraph in [Rat92].

Proposition 5.19. *Let* $f\colon \mathrm{On} \to \mathrm{On}, \xi \mapsto \omega^\xi$.

(a) $f\binom{1}{\omega}$ *denotes the small Veblen ordinal.*

(b) $f\binom{1}{\omega} = \vartheta\Omega^\omega$ *and* $\varphi(\omega, 0) = f\binom{\omega}{1} = \vartheta(\Omega \cdot \omega)$.

Proof. (a) follows essentially from Definition 5.3, Lemma 5.12, and Proposition 5.11. (b) is due to [Sch92], see also [Buc16] or [Ran15]. □

5.2. Fundamental Sequences

As described in the appendix of this article and carried out in [Ran15] with proofs in detail, it is possible to define a primitive recursive ordinal notation system

$$(\mathrm{OT}, \prec)$$

for the small Veblen ordinal that is based on the finitary Veblen functions and whose basic properties (such as being a linear ordering) can be formalized and verified within TID. In particular, fundamental sequences $a[x]$ for ordinal notations $a \in \mathrm{OT}$ and natural numbers $n \in \mathbb{N}$ can be defined with the following properties (see also Subsection A.4).[9]

[9] Suc and Lim in Proposition 5.20 are primitive recursively defined from OT and consist of successor codes and limit codes, respectively. See Definition A.11 in the appendix.

Proposition 5.20.

(a) $\mathsf{TID} \vdash \forall d, x(d \in \mathrm{Suc} \to d[x] \prec d)$.

(b) $\mathsf{TID} \vdash \forall d, x(d \in \mathrm{Lim} \to (0 \prec d[x] \wedge d[x] \prec d[x +_{\mathbb{N}} 1] \wedge d[x] \prec d))$.

(c) $\mathsf{TID} \vdash \forall d, d_0(d \in \mathrm{Lim} \wedge d_0 \prec d \to \exists x(d_0 \prec d[x]))$.

More precisely, the fundamental sequences' definition as given in Definition A.17 in the appendix can be motivated from the set-theoretic perspective of Subsection 5.1. As a consequence, the following interplay of the finitary Veblen functions with the given fundamental sequences can be intuitively understood without relying on the intrinsic properties of the ordinal notation system $(\mathrm{OT}, \prec)$ from Subsection A.1 in the appendix.[10]

Proposition 5.21. *Let $k, m \in \mathbb{N}$. For every $\bar{a}^{(m)}, b, d_0 \in \mathrm{OT}$ with*

$$d_0 \prec \varphi(\bar{a}^{(m+1)}, \bar{0}^{(k)}, b)$$

the following holds over TID*:*

(a) $b \in \mathrm{Lim} \to \exists x\big(d_0 \prec \varphi(\bar{a}^{(m+1)}, \bar{0}^{(k)}, b[x])\big)$.

(b) $(b \notin \mathrm{Lim} \wedge a_1 = 0 \wedge \ldots \wedge a_{m+1} = 0) \to \exists x\big(d_0 \prec \omega^{b[x]} \cdot (x +_{\mathbb{N}} 1)\big)$.

(c) $(b = 0 \wedge a_{m+1} \in \mathrm{Lim}) \to \exists x\big(d_0 \prec \varphi(\bar{a}^{(m)}, a_{m+1}[x], \bar{0}^{(k+1)})\big)$.

However, having a more accessible approach to the wellordering proofs by using fundamental sequences instead of working directly with the underlying ordinal notation system has the cost that one has to verify the fundamental sequences' adequate behaviour in the background (see [Ran15]).

6. Lower Bound $\vartheta\Omega^\omega$ for FIT and TID

In this section, a lower bound for the proof-theoretic ordinal of the theory TID is obtained by means of wellordering proofs. Hence, together with the embedding of TID into FIT from Subsection 4.2, we automatically get a lower bound for FIT as well. We rely for the following proofs on the availability of fundamental sequences for the underlying ordinal notation system $(\mathrm{OT}, \prec)$ as described in Subsection 5.2. This allows us to shift the dependence on the specific implementation of $(\mathrm{OT}, \prec)$ to the properties of the fundamental sequences.

We remark that an alternative approach would be to implement the following proofs directly within the setting of the ordinal notation system $(\mathrm{OT}, \prec)$, avoiding the introduction of fundamental sequences but for the cost of then having the ordinal notation system $(\mathrm{OT}, \prec)$ appear more prominently in the wellordering proofs. In this case, it would be technically more sensible to work with fixed-point free variants $\overline{\varphi}_{n+1}$ of the finitary Veblen functions and base $(\mathrm{OT}, \prec)$ on those (see also [Sch54, §3]). We did not choose this approach for the sake of a better motivation and understanding of the main methods of the wellordering proof. For this section, we fix the following conventions:

[10] $\varphi, \omega, \cdot$ in Proposition 5.21 shall denote the primitive recursive functions $\tilde{\varphi}, \tilde{\omega}, \tilde{\cdot}$, respectively, that simulate the corresponding ordinal arithmetic on the codes based on OT (see Subsection A.2 for details).

(a) We work within TID and presume knowledge on the ordinal notation system $(\mathrm{OT}, \prec)$ that is given in the appendix. See Subsection A.1 for its definition and Subsections A.2 and A.4 for its properties.

(b) The notion *ordinal* denotes terms that are given according to the ordinal notation system $(\mathrm{OT}, \prec)$. Small Greek letters $\alpha, \beta, \gamma, \delta, \ldots$ denote explicit terms for ordinal notations in the sense of OT and which are given externally in the meta-theory. Furthermore, we shall simplify the rigid notation that appears in Subsection A.2 from the appendix by dropping the tilde in the expressions such as $\tilde{\varphi}, \tilde{\omega}, \tilde{+}$ and writing instead $\varphi, \omega, +$, respectively.

(c) P shall denotes $P_\prec$, and analogously Acc, Prog, and TI denote $\mathrm{Acc}_\prec$, $\mathrm{Prog}_\prec$, and $\mathrm{TI}_\prec$, respectively.

Proposition 6.1.

(a) $\mathsf{TID} \vdash \forall x(x \notin \mathrm{OT} \to Px)$.

(b) $\mathsf{TID} \vdash \forall x(Px \to \mathrm{TI}(\mathcal{A}, x))$ *for all* $\mathcal{A} \in \mathcal{L}_{\mathsf{PA}}$.

(c) $\mathsf{TID} \vdash \mathrm{TI}(\mathcal{A}, \alpha)$ *holds for each* $\alpha \prec \omega$ *and* $\mathcal{A} \in \mathcal{L}_{\mathsf{TID}}$.

Proof (Sketch). (a) holds immediately by (CI), using that $a \notin \mathrm{OT}$ implies $b \not\prec a$ for all b. For (b), assume Pa, $\mathrm{Prog}(\mathcal{A})$, and $b \prec a$. We get Pb by (FP) from Remark 4.21, and since $\mathcal{A} \in \mathrm{Pos}_1$ holds, we then get $\mathcal{A}(b)$ by (TID). For (c), note that we can show $\mathrm{TI}(\mathcal{A}, \mathfrak{n}_k)$ for all $k \in \mathbb{N}$ by (meta-)induction on k and where we set $\mathfrak{n}_0 := 0$ and $\mathfrak{n}_{m+1} := \mathfrak{n}_m \tilde{+} \tilde{1}$ for each $m \in \mathbb{N}$. □

Remark 6.2. Dropping the restriction on the induction formula used in (Ind) yields $\mathsf{TID} \vdash \mathrm{TI}(\mathcal{A}, \alpha)$ for each $\alpha \prec \varepsilon_0$ and $\mathcal{A} \in \mathcal{L}_{\mathsf{TID}}$. This is because TID would extend PA in this case with complete induction for the full language $\mathcal{L}_{\mathsf{TID}}$, and we could then follow the usual wellordering proofs for PA and adapt them to the representation of ordinals below ε_0 as given here.

Remark 6.3. Due to Proposition 6.1.(a), we can assume from now on without loss of generality that $a \in \mathrm{OT}$ holds whenever we try to show Pa for some a within TID. In particular, if we aim to prove $P(a+b)$ or $P\varphi(\bar{a}^{(n)})$ for some $a, b, a_1, \ldots, a_n$ (with $n \geq 1$), we shall tacitly assume that $a+b \in \mathrm{OT}$ and $\varphi(\bar{a}^{(n)}) \in \mathrm{OT}$ hold, respectively. Due to Lemma A.12 from the appendix, we then also get $a, b, a_1, \ldots, a_n \in \mathrm{OT}$. Furthermore and since $x \prec y$ implies $x, y \in \mathrm{OT}$ for all x, y by the definition of $\prec$, statements of the form $c \prec a + b$ or $c \prec \varphi(\bar{a}^{(n)})$ imply $a + b \in \mathrm{OT}$ and $\varphi(\bar{a}^{(n)}) \in \mathrm{OT}$, respectively.

Lemma 6.4. $\mathsf{TID} \vdash \forall x, y(Px \land Py \to P(x + y))$.

Proof. Assume a_1, a_2 with Pa_1 and Pa_2, so we have to show $P(a_1 + a_2)$. By showing $\mathrm{Prog}(\mathcal{B})$ for $\mathcal{B} := \Lambda b.P(a_1 + b)$, we can use (TID) together with Pa_2 to get the claim. Now, $\mathrm{Prog}(\mathcal{B})$ is $\forall z(\mathrm{Acc}(\mathcal{B}, z) \to \mathcal{B}(z))$, so assume c and $\mathrm{Acc}(\mathcal{B}, c)$, i.e., $\forall z \prec c(P(a_1 + z))$. Due to (CI), it suffices to show $(\forall z \prec a_1 + c)(Pz)$. Let now $d \prec a_1 + c$. We have either $d \prec a_1$ and can use (FP) on assumption Pa_1 to get Pd, or otherwise, we have $a_1 \preceq d \prec a_1 + c$ by Lemma A.12 from the appendix. In the latter case, we have $d = a_1 + c_0$ for some $c_0 \prec c$, so our assumptions yield the claim. □

6.1. The Simple Case for the Binary Veblen Function

This subsection treats the case for the binary Veblen function separately in order to give a more transparent proof that avoids the technicalities that appear in the treatment of the general case in Subsection 6.2 (e.g., we shall later formulate auxiliary class terms of the form Part_n^k for $1 \leq k \leq n$).

Lemma 6.5. $\mathsf{TID} \vdash \forall x, y(Px \wedge Py \to P\varphi(x, y))$.

Proof. Note that $P0$ and hence $P1$ hold due to (Cl). Now, we assume a_1, a_2 with Pa_1, Pa_2. We use the class term

$$\mathcal{B} := \Lambda a.\forall y(Py \to P\varphi(a, y))$$

with $\mathcal{B} \in \mathrm{Pos}_1$ and show $\mathrm{Prog}(\mathcal{B})$. Then we can use (TID) with Pa_1 and Pa_2. Now, in order to bring the proof of this lemma closer to the proof of Theorem 6.14 that deals with the general case of a finitary Veblen function, we note that $\mathrm{Prog}(\mathcal{B})$ is

$$\forall z(\forall x \prec z(\mathcal{B}(x)) \to \mathcal{B}(z))$$

Now, using the class term

$$\mathcal{A}_2^1 := \Lambda a.\forall y(Py \to \forall x \prec a(P\varphi(x, y)))$$

and that $\mathcal{A}_2^1(a)$ is logically equivalent to $\forall x \prec a(\mathcal{B}(x))$, we get that $\mathrm{Prog}(\mathcal{B})$ is logically equivalent to

$$\forall z(\mathcal{A}_2^1(z) \to \mathcal{B}(z)) \tag{$*$}$$

So, it rests to show $(*)$. For proving this, assume a with

$$\mathcal{A}_2^1(a) \tag{8}$$

and show $\mathcal{B}(a)$, while for proving $\mathcal{B}(a)$, assume b with

$$Pb \tag{9}$$

and show $P\varphi(a, b)$. Once more, we can use (TID), namely with

$$\mathcal{A}_2^2 := \Lambda d.P\varphi(a, d)$$

on (9) since $\mathcal{A}_2^2 \in \mathrm{Pos}_1$ holds, while we have to show $\mathrm{Prog}(\mathcal{A}_2^2)$.[11] Now, for proving $\mathrm{Prog}(\mathcal{A}_2^2)$, we assume d and z with

$$\forall z_0 \prec d(\mathcal{A}_2^2(z_0)) \quad \Big(\text{i.e., } \forall z_0 \prec d(P\varphi(a, z_0))\Big) \tag{10}$$

$$z \prec \varphi(a, d) \tag{11}$$

and show Pz. This yields $P\varphi(a, d)$ by (Cl) because z is arbitrary. We consider now the following case distinction.

[11] Noting our current assumption (8) and our current goal, we remark that we actually show

$$\mathcal{A}_2^1(a) \to \mathrm{Prog}(\mathcal{A}_2^2)$$

which is a special case of Theorem 6.11, and also note that $\mathcal{A}_2^1 \in \mathrm{Pos}_1$ holds with $\mathcal{A}_2^1 \notin \mathrm{Pos}_0$, while we have $\mathcal{A}_2^2 \in \mathrm{Pos}_0$.

1. If $d \in \mathrm{Lim}$: We get that $z \prec \varphi(a, d[x])$ holds for some x by Proposition 5.21. Since we have $d[x] \prec d$ by Proposition 5.20, we get $P\varphi(a, d[x])$ by (10), implying Pz by (FP).
2. If $d \notin \mathrm{Lim}$:
2.1. If $a = 0$: Since $d \notin \mathrm{Lim}$ holds, we get $z \prec t(x)$ for some x by Proposition 5.21, where we let

$$t(x) := \omega^{d[x]} \cdot (x +_{\mathbb{N}} 1)$$

We show $\forall x(P(t(x)))$ by induction on x and note that (Ind) is applicable here because of $\Lambda x.P(t(x)) \in \mathrm{Pos}_1$. For $x = 0$, we can argue as for the case $d \in \mathrm{Lim}$ and get $P(\omega^{d[0]})$. For $x = x_0 +_{\mathbb{N}} 1$, the claim follows from $P(t(0))$, the induction hypothesis, and Lemma 6.4, noting that $d[0] = d[x_0]$ holds by definition and because of $d \notin \mathrm{Lim}$.
2.2. If $a \in \mathrm{Lim}$ and $d = 0$: We have by Proposition 5.21 that $z \prec \varphi(a[x], 0)$ holds for some x. Since we have $a[x] \prec a$ by Proposition 5.20, we get $P\varphi(a[x], 0)$ with (8).
2.3. Otherwise, i.e., either $d = 0$ with $a \in \mathrm{Suc}$ or $d \in \mathrm{Suc}$ with $a \neq 0$: Letting $t := \varphi(a, d)$, we have by Proposition 5.20 some x such that $z \prec t[x]$ holds. Proving

$$\forall x(P(t[x]))$$

by induction on x suffices now. Note again that (Ind) is applicable due to $\Lambda x.P(t[x]) \in \mathrm{Pos}_1$, and note for the following computations of $t[x]$ also that we have $\varphi(a, d) = \phi a d$ by Lemma A.12.
2.3.1. If $x = 0$: If $d = 0$ holds, then we have $t[0] = 1$ and are done since we have $P1$. If $d \in \mathrm{Suc}$ holds with $d = d_0 + 1$, then we have $t[0] = \varphi(a, d_0) + 1$, and since $d_0 \prec d$ holds, we get $P(t[0])$ from (10) and Lemma 6.4 by using $P1$.
2.3.2. If $x = x_0 +_{\mathbb{N}} 1$: We have $t[x_0 +_{\mathbb{N}} 1] = \varphi(a[x_0], t[x_0])$, so the claim follows with $a[x_0] \prec a$ from Proposition 5.20, the induction hypothesis $P(t[x_0])$, and (8). □

Corollary 6.6. $\mathsf{TID} \vdash P\varphi(\alpha, 0)$ *holds for each* $\alpha \prec \omega$.

Proof. The claim is a direct consequence of Lemma 6.5. Note hereby that for $\alpha \prec \omega$, we get $P\alpha$ from Proposition 6.1.(c): We get $\forall x \prec \alpha(Px)$ from $\mathrm{TI}(\Lambda a.Pa, \alpha)$ and closure (Cl), hence $P\alpha$ by (FP). □

Remark 6.7. We proved Lemma 6.5 by applying (TID) to a class term $\mathcal{B}$ in Pos_1 that is not in Pos_0. Though, in order to show $\mathrm{Prog}(\mathcal{B})$ in the proof of Lemma 6.5, we can work with a (weaker) subtheory TID_0 of TID that is obtained from TID by restricting (the instances of) the axiom schemes (TID) and (Ind) to class terms that are in Pos_0 (rather than Pos_1), see [Ran15] for details. This theory TID_0 is the restriction of the theory $\mathsf{ID}_1^*\!\restriction$ to accessible part positive operator forms, i.e., to the language $\mathcal{L}_{\mathsf{TID}}$, while $\mathsf{ID}_1^*\!\restriction$ is a subtheory of ID_1 for positive induction and with the same restriction for complete induction. The proof-theoretic ordinal of $\mathsf{ID}_1^*\!\restriction$ is $\varphi(\omega, 0)$. See for instance [Pro06], and note furthermore Remark 6.13 below.

6.2. The General Case for the Finitary Veblen Functions

Remark 6.8. Note that the expression $\varphi(\bar{a}^{(n+1)})$, i.e., $\tilde{\varphi}(\bar{a}^{(n+1)})$, in Subsection A.2 from the appendix is also defined in case of $n = 0$. We then have $\varphi(a_1) = \omega^{a_1}$.

Definition 6.9. For $k, n \in \mathbb{N}$ with $1 \leq k < n$, we define

$$\begin{aligned}
\mathrm{Part}_n^k &:= \Lambda \bar{a}^{(k)}.\forall y(Py \to \forall x \prec a_k(P\varphi(\bar{a}^{(k-1)}, x, y, \bar{0}^{(n-k-1)}))) \\
\mathrm{Hyp}_n^k &:= \Lambda \bar{a}^{(k)}.\mathrm{Part}_n^1(a_1) \wedge \ldots \wedge \mathrm{Part}_n^k(\bar{a}^{(k)}) \\
\mathrm{Hyp}_n^0 &:= (0 = 0)
\end{aligned}$$

Lemma 6.10. *For $k, n \in \mathbb{N}$ and variables $a_1, \ldots, a_{n-1}$, the following holds:*

(a) $(\Lambda a.P(\varphi(\bar{a}^{(n-1)}, a))) \in \mathrm{Pos}_0$ *for* $1 \leq n$.

(b) $(\Lambda a.\mathrm{Part}_n^k(\bar{a}^{(k-1)}, a)) \in \mathrm{Pos}_1$ *for* $1 \leq k < n$.

Proof. (a) is obvious. For (b), note in the definition of $\mathrm{Part}_n^k(\bar{a}^{(k-1)}, a)$ that Py and $\forall x(x \prec a \to P\varphi(\bar{a}^{(k-1)}, x, y, \bar{0}^{(n-k-1)}))$ are in Pos_0. Furthermore, Py does not contain a as a free variable, so we get indeed that $\forall y(Py \to \forall x \prec a(P\varphi(\bar{a}^{(k-1)}, x, y, \bar{0}^{(n-k-1)})))$ is in $\mathrm{Pos}_1(a)$. $\square$

Theorem 6.11. *For $n \in \mathbb{N}$ with $n \geq 1$, we have*

$$\mathsf{TID} \vdash \forall \bar{a}^{(n-1)}\big(\mathrm{Hyp}_n^{n-1}(\bar{a}^{(n-1)}) \to \mathrm{Prog}(\Lambda a.P\varphi(\bar{a}^{(n-1)}, a))\big)$$

Proof. Let $n \geq 1$ and $\bar{a}^{(n-1)}$ be given with

$$\mathrm{Hyp}_n^{n-1}(\bar{a}^{(n-1)}) \tag{12}$$

In order to show $\mathrm{Prog}(\Lambda a.P\varphi(\bar{a}^{(n-1)}, a))$, assume a and d such that

$$\forall x \prec a(P\varphi(\bar{a}^{(n-1)}, x)) \tag{13}$$

$$d \prec \varphi(\bar{a}^{(n-1)}, a) \tag{14}$$

hold and show Pd. This would yield $P\varphi(\bar{a}^{(n-1)}, a)$ by (CI) because d is arbitrary.
<u>1.</u> If $n = 1$ or $a_1 = \ldots = a_{n-1} = 0$ hold: We can proceed as in Lemma 6.5 since we have $\varphi(\bar{a}^{(n-1)}, a) = \varphi(a) = \omega^a$.
<u>2.</u> Otherwise: We can assume now that some $1 \leq l \leq n-1$ exists with

$$a_l \neq 0 \ \ \& \ \ a_{l+1} = \ldots = a_{n-1} = 0$$

i.e., that we have $\varphi(\bar{a}^{(n-1)}, a) = \varphi(\bar{a}^{(l)}, \bar{0}^{(n-l-1)}, a)$ with $a_l \neq 0$. Furthermore, (12) yields

$$\mathrm{Part}_n^1(a_1) \wedge \ldots \wedge \mathrm{Part}_n^{n-1}(\bar{a}^{(n-1)}) \tag{15}$$

Consider now the following case distinction and note that $P0$ and hence $P1$ hold due to (CI).
<u>2.1.</u> If $a \in \mathrm{Lim}$: We get that $d \prec \varphi(\bar{a}^{(n-1)}, a[x])$ holds for some x by Proposition 5.21 and (14). Since we have $a[x] \prec a$ by Proposition 5.20, we get $P\varphi(\bar{a}^{(n-1)}, a[x])$ by (13) which implies Pd by (FP).
<u>2.2.</u> If $a \notin \mathrm{Lim}$:
<u>2.2.1.</u> If $a_l \in \mathrm{Lim}$ and $a = 0$: By Proposition 5.21, we have some x such that $z \prec \varphi(\bar{a}^{(l-1)}, a_l[x], \bar{0}^{(n-l)})$ holds. We get $P\varphi(\bar{a}^{(l-1)}, a_l[x], \bar{0}^{(n-l)})$ with $\mathrm{Part}_n^l(\bar{a}^{(l)})$ from (15) because we have $a_l[x] \prec a_l$ by Proposition 5.20.

2.2.2. Otherwise, i.e., either $a = 0$ with $a_l \in \mathrm{Suc}$ or $a \in \mathrm{Suc}$ with $a_l \neq 0$: In this situation, Lemma A.12 implies $\varphi(\bar{a}^{(l)}, \bar{0}^{(n-l)}) = \phi a_p \dots a_l \bar{0}^{(n-l-1)} a$ for some $1 \leq p \leq l$ where $a_1, \dots, a_p = 0$ holds. Bear in mind that ϕ is from the ordinal notation system $(\mathrm{OT}, \prec)$ in the appendix. In order to simplify notation and without loss of generality, we shall assume $p = 1$, noting that the following argument works for the general case as well. Letting

$$t := \phi \bar{a}^{(l)} \bar{0}^{(n-l-1)} a \tag{16}$$

we have by Proposition 5.20 some x such that $z \prec t[x]$ holds. Proving

$$\forall x (P(t[x]))$$

by induction on x suffices now. (Ind) is applicable because $\Lambda x.P(t[x]) \in \mathrm{Pos}_1$ holds.

2.2.2.1. If $x = 0$: For $a = 0$, we have $t[0] = 1$ and are done since we have $P1$. If $a \in \mathrm{Suc}$ holds with $a = a_0 + 1$, then we have $t[0] = \varphi(\bar{a}^{(n-1)}, a_0) + 1$ due to the form of t in (16) and the definition of $t[0]$. Since $a_0 \prec a$ holds, we get $P(t[0])$ from (13) and Lemma 6.4 by using $P1$.

2.2.2.2. If $x = x_0 +_{\mathbb{N}} 1$: We get $t[x_0 +_{\mathbb{N}} 1] = \varphi(\bar{a}^{(l-1)}, a_l[x_0], t[x_0], \bar{0}^{(n-l)})$, so the claim follows with $a_l[x_0] \prec a_l$ from Proposition 5.20, the induction hypothesis $P(t[x_0])$, and $\mathrm{Part}^l_n(\bar{a}^{(l)})$ from (15). □

Corollary 6.12. *For $n \in \mathbb{N}$ with $n \geq 1$, we have*

$$\mathsf{TID} \vdash \forall \bar{a}^{(n)} \big(\mathrm{Hyp}^{n-1}_n(\bar{a}^{(n-1)}) \wedge P a_n \to P\varphi(\bar{a}^{(n)}) \big)$$

Proof. Immediate from Theorem 6.11 by using (TID) and Lemma 6.10.(a). □

Remark 6.13. Note that we did not invoke (TID) in the proof of Theorem 6.11, so this result holds also for the restriction TID_0 of TID that we mentioned in Remark 6.7. Clearly, our proof of Theorem 6.11 does not work directly within PA because we invoked (CI) and (FP).

Theorem 6.14. *For $k, n \in \mathbb{N}$ with $1 \leq k < n$, we have*

$$\mathsf{TID} \vdash \forall \bar{a}^{(k-1)} \big(\mathrm{Hyp}^{k-1}_n(\bar{a}^{(k-1)}) \to \mathrm{Prog}(\Lambda a.\mathrm{Part}^k_n(\bar{a}^{(k-1)}, a)) \big)$$

Proof. We fix $n \geq 1$ and argue by induction on $n - k$ for $1 \leq k < n$. Let $\bar{a}^{(k-1)}$ be given with

$$\mathrm{Hyp}^{k-1}_n(\bar{a}^{(k-1)}) \tag{17}$$

and where (17) just gives us the formula $0 = 0$ in case of $k = 1$. Assume now a, a_k, a_{k+1} with

$$\forall x \prec a (\mathrm{Part}^k_n(\bar{a}^{(k-1)}, x)) \tag{18}$$

$$P a_{k+1} \tag{19}$$

$$a_k \prec a \tag{20}$$

in order to show $\mathrm{Prog}(\Lambda a.\mathrm{Part}^k_n(\bar{a}^{(k-1)}, a))$. In case we have $k \neq n - 1$, further let

$$a_i := 0$$

for each $k < i \leq n$. We have to show $P(\varphi(\bar{a}^{(k+1)}, \bar{0}^{(n-k-1)}))$, i.e.,

$$P(\varphi(\bar{a}^{(n)})) \tag{$*$}$$

From (20) and (18), we get

$$\mathrm{Part}_n^k(\bar{a}^{(k-1)}, a_k) \tag{21}$$

and hence

$$\mathrm{Hyp}_n^k(\bar{a}^{(k)}) \tag{22}$$

with (17). From (19) and $P0$, we get

$$Pa_{k+1} \wedge \ldots \wedge Pa_n \tag{23}$$

We show by a side induction on i that the following holds:

$$1 \leq i < n \quad \Longrightarrow \quad \mathrm{Hyp}_n^i(\bar{a}^{(i)}) \tag{$**$}$$

From $(**)$ with $i := n-1$ and Pa_n from (23), we then get $(*)$ by Corollary 6.12. For the proof of $(**)$, we note that the claim follows in case of $1 \leq i \leq k$ from (22). If we have $k < i < n$, then we can use the side induction hypothesis and get

$$\mathrm{Hyp}_n^{i-1}(\bar{a}^{(i-1)}) \tag{24}$$

This and the main induction hypothesis yield $\mathrm{Prog}(\Lambda a.\mathrm{Part}_n^i(\bar{a}^{(i-1)}, a))$ and hence we get $\forall a(Pa \rightarrow \mathrm{Part}_n^i(\bar{a}^{(i-1)}, a))$ by (TID), while noting here Lemma 6.10.(b). Now, $\mathrm{Part}_n^i(\bar{a}^{(i)})$ follows from (23) and the current case $k < i < n$. Hence, we get $\mathrm{Hyp}_n^i(\bar{a}^{(i)})$ by (24). □

Corollary 6.15. *For $k, n \in \mathbb{N}$ with $1 \leq k < n$, we have*

$$\mathsf{TID} \vdash \forall \bar{a}^{(k)} \big(\mathrm{Hyp}_n^{k-1}(\bar{a}^{(k-1)}) \wedge Pa_k \rightarrow \mathrm{Hyp}_n^k(\bar{a}^{(k)})\big)$$

Proof. From $\mathrm{Hyp}_n^{k-1}(\bar{a}^{(k-1)}) \wedge Pa_k$, we get $\mathrm{Part}_n^k(\bar{a}^{(k)})$ by Theorem 6.14 and (TID), while noting Lemma 6.10.(b). Hence, we get $\mathrm{Hyp}_n^k(\bar{a}^{(k)})$. □

Theorem 6.16. *For each $n \geq 1$, we have*

$$\mathsf{TID} \vdash \forall \bar{a}^{(n)} (\textstyle\bigwedge_{i=1}^n Pa_i \rightarrow P\varphi(\bar{a}^{(n)}))$$

Proof. Let $n \geq 1$ and assume $\bar{a}^{(n)}$ with $\bigwedge_{i=1}^n Pa_i$. We trivially get

$$\mathrm{Hyp}_n^0(\bar{a}^{(0)}) \wedge \textstyle\bigwedge_{i=1}^n Pa_i \tag{25}$$

due to the definition of Hyp_n^0. We now show by induction on $k \in \mathbb{N}$ that the following holds:

$$0 \leq k < n \implies \mathrm{Hyp}_n^k(\bar{a}^{(k)}) \wedge \textstyle\bigwedge_{i=k+1}^n Pa_i \tag{$*$}$$

Then the claim $P\varphi(\bar{a}^{(n)})$ follows from Corollary 6.12 and $(*)$ with $k := n-1$. We show now $(*)$ and assume $0 \leq k < n$:
<u>1.</u> $k = 0$: This is (25).
<u>2.</u> $0 < k \leq n$: The induction hypothesis yields $\mathrm{Hyp}_n^{k-1}(\bar{a}^{(k-1)}) \wedge \bigwedge_{i=k}^n Pa_i$ and hence the claim $(*)$ due to Corollary 6.15. □

Corollary 6.17 (Lower bound of TID)**.** *For each $\mathcal{A} \in \mathcal{L}_{\mathsf{PA}}$ and $\alpha \in \mathrm{OT}$, we have*

$$\mathsf{TID} \vdash \mathrm{TI}(\mathcal{A}, \alpha)$$

Proof. By induction on the build-up of $\alpha \in \mathrm{OT}$. We can use Lemma 6.4 and Theorem 6.16 together with Proposition 6.1.(b). □

7. Upper Bound $\vartheta\Omega^\omega$ for FIT and TID

For determining the upper bound of FIT, we apply one result from [JS99] that relates over ACA_0 the scheme (Π^1_3-RFN) of ω-model reflection for Π^1_3 formulas to the scheme (Π^1_2-BI) of bar induction for Π^1_2 formulas, and one result of [RW93] that determines the proof-theoretic ordinal of Π^1_2-BI_0 to be the small Veblen ordinal $\vartheta\Omega^\omega$. Then an embedding of FIT into the second order theory Π^1_3-RFN_0 of ω-model reflection for Π^1_3 formulas suffices to get the desired upper bound result for FIT. Moreover and due to Section 4.2, this also provides an upper bound for TID. In particular, we shall exploit the Π^1_1 definability of a least fixed-point. A similar approach has been taken in [AR10] and [Pro06] for the treatment of the theories Π^1_2-RFN_0 and ID^*_1 (a subsystem of ID_1 that allows only positive induction for the predicates $P_\mathfrak{A}$ that are assigned to each positive operator form $\mathfrak{A}$). Below, we shall provide an upper bound for FIT by embedding it directly into Π^1_3-RFN_0. We remark that if we were to investigate only the subtheory TID of ID_1, we could have embedded it directly into Π^1_3-RFN_0 (rather than taking the detour via FIT as figure 1 from the introduction on page 3 suggests). Furthermore, we recall that D. Probst's *modular ordinal analysis* from [Pro16] determines the proof-theoretic ordinal of Π^1_3-RFN_0 to be the small Veblen ordinal by *metapredicative* methods.

7.1. Subsystems of Second Order Arithmetic

We shall introduce here subsystems of second order arithmetic, and we formulate them in the language $\mathcal{L}^2_{\mathsf{PA}}$ that we defined in Section 4. In particular, recall that $\mathcal{L}^2_{\mathsf{PA}}$ formulas allow for quantification over set variables X. The following definitions are taken to some extent from [JS99] and [Sim09], respectively, and we refer to these sources for more details on subsystems of second order arithmetic and in particular to the underlying two-sorted logic.

Definition 7.1. We use the following standard abbreviations

$$\begin{aligned} (X)_t &:= \Lambda a.\langle t,a\rangle \in X && \\ (QY \mathrel{\dot\in} X)A &:= (Qy)A((X)_y/Y) && \text{(where } Q\in\{\forall,\exists\}) \\ Y \mathrel{\dot\in} X &:= (\exists Z \mathrel{\dot\in} X)(Z=Y) && \text{(i.e., } Y \mathrel{\dot\in} X \text{ is } \exists z((X)_z = Y)) \end{aligned}$$

and we define the relativization A^X of a formula A to a set variable X inductively as follows:

$$\begin{array}{ll} A & \text{if } A \text{ is an atomic formula} \\ \neg(A_0^X) & \text{if } A \text{ is } \neg A_0 \\ A_0^X \circ A_1^X & \text{if } A \text{ is } A_0 \circ A_1 \text{ and } \circ \in \{\vee,\wedge,\rightarrow\} \\ (Qx)A_0^X & \text{if } A \text{ is } (Qx)A_0 \text{ and } Q\in\{\exists,\forall\} \\ (QY \mathrel{\dot\in} X)A_0^X & \text{if } A \text{ is } (QX)A_0 \text{ and } Q\in\{\exists,\forall\} \end{array}$$

As usual, we assume tacitly a renaming of bound variables in order to avoid a clash of variables. Note that set variables occur at most free in A^X, i.e., A^X is arithmetical.

Notation 7.2. We also write

$$X \models A$$

in order to denote A^X.

Definition 7.3 (Usual hierarchies of formulas)**.**

(a) Π^1_0 *(or also* Σ^1_0*) formulas* are called those formulas A that are arithmetical, i.e., $\mathcal{L}^2_{\mathsf{PA}}$ formulas without quantifications over set variables. We denote this also by writing $A \in \Pi^1_0$ or $A \in \Sigma^1_0$.

(b) Π^1_{n+1} *formulas* are called those formulas which are of the form

$$\forall X_1 \exists X_2 \ldots (Q_{n+1} X_{n+1}) A$$

for some $A \in \Pi^1_0$, and where Q_{n+1} is $\exists$ for even n and Q_{n+1} is $\forall$ otherwise. We denote this also by writing $A \in \Pi^1_n$.

(c) Σ^1_{n+1} *formulas* are all those formulas which are of the form $\exists X A$ with $A \in \Pi^1_n$. We denote this also by writing $A \in \Sigma^1_n$.

Definition 7.4. The two-sorted theory ACA_0 is based on the language $\mathcal{L}^2_{\mathsf{PA}}$. Its axioms are the axioms of PA without complete induction, and where the equality axioms (for the first sort) hold for the language $\mathcal{L}^2_{\mathsf{PA}}$. Moreover, ACA_0 consists of the following principles:

- *Set induction*:

$$\forall X((0 \in X \land \forall x(x \in X \to \mathbf{S}x \in X)) \to \forall x(x \in X))$$

- *Arithmetical comprehension*:

$$(\mathsf{ACA}) \quad \exists X \forall x(x \in X \leftrightarrow A)$$

  for each $A \in \Pi^1_0$ that does not contain X (though it might contain free occurrences of other set variables).

Proposition 7.5. ACA_0 *is finitely axiomatizable by a* Π^1_2*-sentence* $\mathrm{F}_{\mathsf{ACA}}$.

Proof. See for instance [Sim09, Lemma VIII.1.5]. □

Definition 7.6. We define the following principles:

- Σ^1_1 *axiom of choice*:

$$(\Sigma^1_1\text{-}\mathsf{AC}) \quad \forall x \exists X A \to \exists Y \forall x(A((Y)_x/X))$$

  for each $A \in \Sigma^1_1$.

- Σ^1_1 *axiom of dependent choice*:

$$(\Sigma^1_1\text{-}\mathsf{DC}) \quad \begin{cases} \forall x \forall X \exists Y A \\ \to \forall U \exists Z\big((Z)_0 = U \land \forall x(A((Z)_x/X, (Z)_{x+1}/Y))\big) \end{cases}$$

  for each $A \in \Sigma^1_1$.

- Π^1_n *ω-model reflection* for $n \in \mathbb{N}$:

$$(\Pi^1_n\text{-}\mathsf{RFN}) \quad \begin{cases} \forall U_1, \ldots, U_k \\ \big(A \to \exists X(A^X \wedge \mathrm{F}^X_{\mathsf{ACA}} \wedge U_1 \mathrel{\dot\in} X \wedge \ldots \wedge U_k \mathrel{\dot\in} X)\big) \end{cases}$$

  for each $A \in \Pi^1_n$ with at most $U_1, \ldots, U_k$ occurring as free set variables in A (and where $\mathrm{F}_{\mathsf{ACA}}$ is taken from Proposition 7.5).

- Π^1_n *bar induction* for $n \in \mathbb{N}$:

$$(\Pi^1_n\text{-}\mathsf{BI}) \quad \forall X(\mathrm{WO}(X) \to \mathrm{TI}_X(\Lambda a.A))$$

  for each $A \in \Pi^1_n$ and where we let

$$\begin{aligned} \mathrm{WO}(X) &:= \mathrm{LO}(X) \wedge \mathrm{WF}(X) \\ \mathrm{WF}(X) &:= \forall Y(\mathrm{TI}_X(Y)) \\ \mathrm{TI}_X(\Lambda a.A) &:= \mathrm{PROG}(X, \Lambda a.A) \to \forall x A(x/a) \\ \mathrm{PROG}(X, \Lambda a.A) &:= \forall x(\forall y\big(\langle y, x\rangle \in X \to A(y/a)\big) \to A(x/a)) \end{aligned}$$

  and where $\mathrm{LO}(X)$ denotes the usual arithmetical formula that expresses that X encodes a binary relation that is a linear ordering.

The theories $\Sigma^1_1\text{-}\mathsf{AC}_0$, $\Sigma^1_1\text{-}\mathsf{DC}_0$, $\Pi^1_n\text{-}\mathsf{RFN}_0$, and $\Pi^1_n\text{-}\mathsf{BI}_0$ are defined by extending ACA_0 with the axiom scheme (Σ^1_1-AC), (Σ^1_1-DC), (Π^1_n-RFN), and (Π^1_n-BI), respectively.

Remark 7.7. We added the definition for (Π^1_n-BI) for the sake of completeness but we shall not need to use it directly in the following.

7.2. Upper Bound Results from the Literature

Theorem 7.8 ([RW93])**.** *$|\Pi^1_{n+2}\text{-}\mathsf{BI}_0| = \vartheta\Omega(n+1, \omega)$ holds for all $n \in \mathbb{N}$.*

Theorem 7.9 ([JS99])**.** *(Π^1_{n+1}-BI) and (Π^1_{n+2}-RFN) are equivalent over ACA_0 for all $n \in \mathbb{N}$.*

Theorem 7.10 ([Sim09])**.** *Over ACA_0, we have*

(a) (Π^1_n-RFN) *implies* (Π^1_k-RFN) *for $k \leq n$.*

(b) (Π^1_2-RFN) *is equivalent to* (Σ^1_1-DC).

(c) (Σ^1_1-DC) *implies* (Σ^1_1-AC).

Corollary 7.11. (Π^1_{n+2}-RFN) *implies* (Σ^1_1-AC) *over ACA_0 for all $n \in \mathbb{N}$.*

Theorem 7.12 ([Can86])**.** $|\Sigma^1_1\text{-}\mathsf{DC}_0| = \varphi(\omega, 0)$.

Corollary 7.13. *$|\Pi^1_{n+2}\text{-}\mathsf{RFN}_0| = \vartheta\Omega(n, \omega)$ holds for all $n \in \mathbb{N}$.*

Proof. For $n \geq 1$, this is immediate from Theorem 7.8 and Theorem 7.9. For $n = 0$, use also Theorem 7.12 and that $\vartheta\Omega(0, \omega) = \vartheta(\Omega \cdot \omega) = \varphi(\omega, 0)$ holds by Proposition 5.19.(b). □

7.3. Some Syntactical Properties of $\mathcal{L}^2_{\mathsf{PA}}$ Formulas

Definition 7.14 (Refined hierarchies of formulas)**.** Let T be some theory of $\mathcal{L}^2_{\mathsf{PA}}$ as introduced in Subsection 7.1, e.g., $\mathsf{T} = \mathsf{ACA}_0$ or $\mathsf{T} = \Sigma^1_1\text{-}\mathsf{AC}_0$.

(a) Π^1_n *formulas over* T are all $A \in \mathcal{L}^2_{\mathsf{PA}}$ that are provably equivalent over T to some formula $A' \in \Pi^1_n$.

(b) $A \in \Pi^1_n(\mathsf{T})$ denotes that A is an Π^1_n formula over T.

Remark 7.15. In case that $A \in \Pi^1_n(\mathsf{T})$ is given for some theory T of $\mathcal{L}^2_{\mathsf{PA}}$ and we consider some $A' \in \Pi^1_n$ that is provably equivalent over T to A, then we can assume that A and A' have the same free variables, and we shall tacitly do so from now on. Moreover, if $\mathsf{T}_1, \mathsf{T}_2$ are theories of $\mathcal{L}^2_{\mathsf{PA}}$ as introduced in Subsection 7.1 such that T_2 comprises T_1, then obviously $A \in \Pi_n(\mathsf{T}_1)$ implies $A \in \Pi_n(\mathsf{T}_2)$.

Proposition 7.16. *Let $k, n \in \mathbb{N}$ and $\mathsf{T} \in \{\mathsf{ACA}_0, \Sigma^1_1\text{-}\mathsf{AC}_0\}$. Then we have the following.*

(a) $(A \in \Pi^1_k(\mathsf{T}) \;\;\&\;\; k < n) \implies (A \in \Pi^1_n(\mathsf{T}) \;\;\&\;\; \neg A \in \Pi^1_n(\mathsf{T}))$.

(b) $\Pi^1_n(\mathsf{T})$ *is closed under conjunction, disjunction, and universal quantification for number variables, i.e., we have*

$$\begin{aligned} A, B \in \Pi^1_n(\mathsf{T}) &\implies A \circ B \in \Pi^1_n(\mathsf{T}) \quad \textit{where } \circ \in \{\wedge, \vee\} \\ A \in \Pi^1_n(\mathsf{T}) &\implies \forall x A \in \Pi^1_n(\mathsf{T}) \end{aligned}$$

(c) $\Pi^1_{n+1}(\mathsf{T})$ *is closed under universal quantification for set variables, i.e., we have*

$$A \in \Pi^1_{n+1}(\mathsf{T}) \implies \forall X A \in \Pi^1_{n+1}(\mathsf{T})$$

Proof. Straightforward by using essentially (ACA), see [Ran15]. □

Corollary 7.17. *Let $k, n \in \mathbb{N}$ and $\mathsf{T} \in \{\mathsf{ACA}_0, \Sigma^1_1\text{-}\mathsf{AC}_0\}$. Then we have*

$$\begin{aligned} &\big(A_0, \ldots, A_k \in \Pi^1_n(\mathsf{T}) \;\;\&\;\; B \in \Pi^1_{n+1}(\mathsf{T})\big) \\ &\quad \implies \forall \vec{x}(A_0 \to \ldots \to A_k \to B) \in \Pi^1_{n+1}(\mathsf{T}) \end{aligned}$$

Proof. Immediate by Proposition 7.16 and induction on $k \in \mathbb{N}$, while noting that $A_k \in \Pi^1_n(\mathsf{T})$ implies $\neg A_k \in \Pi^1_{n+1}(\mathsf{T})$, and that $A_k \to B$ is equivalent to $\neg A_k \vee B$. □

7.4. Embedding FIT into $\Pi^1_3\text{-}\mathsf{RFN}_0$

In order to interpret within $\Pi^1_3\text{-}\mathsf{RFN}_0$ the applicative part of FIT, i.e., **I.** in Definition 3.12, we shall first implement the so-called *canonical model* $\mathfrak{PR}$ for this applicative part. It is built upon ordinary recursion theory and by using indices of *partial recursive* functions for interpreting the function symbol $\cdot$ of $\mathcal{L}_{\mathsf{FIT}}$. For a thorough introduction to this construction and a more detailed treatment of the following (in a slightly different setting), we refer to [FJS]. Without going into detail, we let $\mathbf{T}$ be the ternary, primitive recursive relation $\mathbf{T}$ according to Kleene's Normal Form Theorem, and let $\mathbf{U}$ be the corresponding unary primitive recursive (result-extracting) function, and in the sense that $\exists x(\mathbf{T}(e, \langle n_1, \ldots, n_k \rangle, x) \wedge \mathbf{U}(x) = m)$ for

$e, k, m, n_1, \ldots, n_k \in \mathbb{N}$ corresponds to the expression $\{e\}(n_1, \ldots, n_k) \simeq m$ in the usual sense that $\{e\}$ denotes the partial recursive function indexed by the number e. Furthermore, let $\mathsf{\Pi}^1_1(x, y)$ with $x \neq y$ denote a universal Π^1_1 formula for Π^1_1 formulas that have one free variable, i.e., we have $\mathsf{\Pi}^1_1(x, y) \in \Pi^1_1$ and for each $\mathcal{L}^2_{\mathsf{PA}}$ formula $A \in \Pi^1_1$ with $\mathrm{FV}(A) = \{y\}$, we have that $\exists x \forall y(\mathsf{\Pi}^1_1(x, y) \leftrightarrow A)$ holds over ACA_0.[12]

Definition 7.18 (Interpretation of $\mathcal{L}_{\mathsf{FIT}}$ into $\mathcal{L}^2_{\mathsf{PA}}$)**.** In the abovementioned setting, we let $\mathbf{T}$ and $\mathbf{U}$ also denote the corresponding relation and function symbols in $\mathcal{L}_{\mathsf{PA}}$, and then we set

$$(\{a\}(b) \simeq c) := \exists x(\mathbf{T}(a, b, x) \wedge \mathbf{U}(x) = c)$$

Next, we assume an assignment of the constants k, s of $\mathcal{L}_{\mathsf{FIT}}$ to numerals $\mathsf{k}^\star, \mathsf{s}^\star$ that have corresponding properties over ACA_0 as described by the axiom group **I.1.** in definition 3.12. For the remaining constants $\mathsf{p}, \mathsf{p}_0, \mathsf{p}_1, 0, \mathsf{s_N}, \mathsf{p_N}, \mathsf{d_N}$ of $\mathcal{L}_{\mathsf{FIT}}$, we set $\mathsf{p}^\star$ to be the numeral of the (primitive recursive) function $(m, n) \mapsto \langle m, n\rangle$; $\mathsf{p}_i^\star$ to be the numeral of $m \mapsto (m)_i$ for $i = 0, 1$; $0^\star$ to be 0; $\mathsf{s}^\star_{\mathsf{N}}$ to be the numeral of $m \mapsto m + 1$; $\mathsf{p}^\star_{\mathsf{N}}$ to be the numeral of $m \mapsto m \dot{-} 1$; and $\mathsf{d}^\star_{\mathsf{N}}$ to be the numeral of the case distinction function, mapping (k, l, m) to l if $k = 0$, otherwise to m. See also [FJS].

The translation $\mathrm{V}^\star_t(x)$ of a $\mathcal{L}_{\mathsf{FIT}}$ term t into the language of $\mathcal{L}_{\mathsf{PA}}$ is defined for variables $x \notin \mathrm{FV}(t)$ as follows:

$$\begin{array}{ll} t = x & \text{if } t \text{ is a variable} \\ t^\star = x & \text{if } t \text{ is a constant} \\ \exists y, z(\mathrm{V}^\star_r(y) \wedge \mathrm{V}^\star_s(z) \wedge \{y\}(z) \simeq x) & \text{if } t \text{ is } rs \end{array}$$

and for each $\mathcal{L}_{\mathsf{FIT}}$ formula A, we let the $\mathcal{L}^2_{\mathsf{PA}}$ formula $A^\star$ be defined recursively on the build-up of A as follows for every $x \notin \mathrm{FV}(A)$ (while we shall provide the case where A is $t \in \mathsf{I}_{\mathbb{P},\mathbb{Q}}$ in ($\star$-$\mathsf{I}_{\mathbb{P},\mathbb{Q}}$) below):

$$\begin{array}{ll} \exists x(\mathrm{V}^\star_s(x) \wedge \mathrm{V}^\star_t(x)) & \text{if } A \text{ is } s = t \\ \exists x(\mathrm{V}^\star_t(x)) & \text{if } A \text{ is } t\downarrow \text{ or } t \in \mathsf{N} \\ \exists x(\mathrm{V}^\star_t(x) \wedge x \in \mathsf{U}) & \text{if } A \text{ is } t \in \mathsf{U} \\ \mathrm{V}^\star_t(0) \wedge \neg \mathrm{V}^\star_t(0) & \text{if } A \text{ is } t \in \overline{\mathsf{N}} \text{ (see Lemma 7.19)} \\ \exists x(\mathrm{V}^\star_t(x) \wedge x \in X) & \text{if } A \text{ is } t \in X \\ \exists x(\mathrm{V}^\star_t(x) \wedge B^\star(z/x)) & \text{if } A \text{ is } t \in \{z\colon B\} \text{ for } B \in \mathrm{For}^+ \\ \neg(B^\star) & \text{if } A \text{ is } \neg B \\ B^\star \circ C^\star & \text{if } A \text{ is } B \circ C \text{ for } \circ \in \{\wedge, \vee, \to\} \\ QzB^\star & \text{if } A \text{ is } QzB \text{ for } Q \in \{\forall, \exists\} \\ Qz(B^\star\big((\Lambda a.\mathsf{\Pi}^1_1(z, a))/X\big)) & \text{if } A \text{ is } QXB \text{ for } Q \in \{\forall, \exists\} \end{array}$$

and for the case that A is $t \in \mathsf{I}_{\mathbb{P},\mathbb{Q}}$, we introduce first the following positive operator form (for any $\mathbb{P}, \mathbb{Q} \in \mathrm{Ty}\!\restriction$)

$$\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}} := \Lambda X \Lambda x.(x \in \mathbb{P})^\star \wedge \forall y((y \in \mathbb{P})^\star \to (\langle y, x\rangle \in \mathbb{Q})^\star \to y \in X)$$

[12] Bear in mind that this universal Π^1_1 formula shall include the unary relation symbol U of $\mathcal{L}_{\mathsf{PA}}$ as a parameter.

and note that $\mathbb{P}, \mathbb{Q}$ do not contain expressions of the form $\mathsf{I}_{\mathbb{P}',\mathbb{Q}'}$. Eventually, we set

$$\big(t \in \mathsf{I}_{\mathbb{P},\mathbb{Q}}\big)^\star := \forall X(\forall x(\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(X,x) \to x \in X) \to (t \in X)^\star) \qquad (\star\text{-}\mathsf{I}_{\mathbb{P},\mathbb{Q}})$$

Furthermore, we tacitly assume in the definition of the translation $A^\star$ as usual a renaming of bound variables in order to avoid a clash of variables. Note also that the translation is meant to interpret type variables as Π^1_1 definable sets and that $\mathsf{I}_{\mathbb{P},\mathbb{Q}} \in \mathrm{Ty}$ implies that $\mathbb{P}, \mathbb{Q}$ do not contain type variables (since $\mathbb{P}, \mathbb{Q} \in \mathrm{Ty}{\restriction}$).

Lemma 7.19. *Let $A \in \mathcal{L}_{\mathsf{FIT}}$, then $A^\star$ and A have the same free variables.*

Proof. This is clear from the definition of $A^\star$, while note that it is due to this lemma that we defined $(t \in \overline{\mathsf{N}})^\star$ as $\mathrm{V}^\star_t(0) \wedge \neg\mathrm{V}^\star_t(0)$ instead as $\neg(0 = 0)$. $\square$

Remark 7.20. For any $\mathbb{F} \in \mathrm{FT}$, consider the $\mathcal{L}^2_{\mathsf{PA}}$ class term $\mathcal{A} := \Lambda z.(tz \in \mathbb{F})^\star$. In order to make later arguments more readable, we shall make the translation of the $\mathcal{L}_{\mathsf{FIT}}$ formula $\mathrm{Cl}_{\mathbb{P},\mathbb{Q}}(\Lambda z.tz \in \mathbb{F})$ more explicit (cf., Notation 3.9):

$$\left.\begin{aligned} &\big(\mathrm{Cl}_{\mathbb{P},\mathbb{Q}}(\Lambda z.tz \in \mathbb{F})\big)^\star \\ &= \begin{pmatrix} \forall x(x \in \mathbb{P} \wedge (\forall y \in \mathbb{P})(\langle y, x\rangle \in \mathbb{Q} \to ty \in \mathbb{F})) \\ \to tx \in \mathbb{F} \end{pmatrix}^\star \\ &= \forall x(\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(\mathcal{A}, x) \to \mathcal{A}(x)) \end{aligned}\right\} \qquad (\star\text{-}\mathrm{Cl}_{\mathbb{P},\mathbb{Q}})$$

As mentioned in Remark 3.10, we defined $\mathrm{Cl}_{\mathbb{P},\mathbb{Q}}(\Lambda z.tz \in \mathbb{F})$ in Section 3 in order to have the above representation that allows to use $\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}$ in a intuitive way. This correspondence would appear as directly as here in case we would have defined $\mathrm{Cl}_{\mathbb{P},\mathbb{Q}}(\Lambda z.tz \in \mathbb{F})$ for instance as

$$\forall x(x \in \mathbb{P} \to (\forall y \in \mathbb{P})(\langle y, x\rangle \in \mathbb{Q} \to ty \in \mathbb{F})) \to tx \in \mathbb{F}$$

Lemma 7.21. *Let $n \geq 0$. For each $\mathcal{L}_{\mathsf{FIT}}$ term t and each $\mathcal{L}_{\mathsf{PA}}$ term r, the following holds.*

(a) $\mathrm{V}^\star_t(x) \in \Pi^1_0$.

(b) $(t \in X)^\star \in \Pi^1_0$ & $\big(\mathbb{P} \in \mathrm{Ty}{\restriction} \implies (t \in \mathbb{P})^\star \in \Pi^1_0\big)$.

(c) For $\mathsf{T} \in \{\mathsf{ACA}_0, \Sigma^1_1\text{-}\mathsf{AC}_0\}$, $\mathcal{B} := \Lambda a.B(a)$ with $B \in \mathcal{L}^2_{\mathsf{PA}}$, and $\mathbb{P}, \mathbb{Q} \in \mathrm{Ty}{\restriction}$, we have:

$$B \in \Pi^1_n(\mathsf{T}) \implies \mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(\mathcal{B}, r) \in \Pi^1_n(\mathsf{T})$$

In particular, we have $\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(\mathcal{B}, r) \in \Pi^1_0$ in case of $B \in \Pi^1_0$.

(d) $\mathbb{P}, \mathbb{Q} \in \mathrm{Ty}{\restriction} \implies \big(\mathrm{Cl}_{\mathbb{P},\mathbb{Q}}(X)^\star \in \Pi^1_0$ & $(t \in \mathsf{I}_{\mathbb{P},\mathbb{Q}})^\star \in \Pi^1_1\big)$.

(e) $A \in \mathrm{For}^+ \implies A^\star \in \Pi^1_1(\Sigma^1_1\text{-}\mathsf{AC}_0)$.

(f) $\mathbb{F} \in \mathrm{FT} \implies (t \in \mathbb{F})^\star \in \Pi^1_2(\Sigma^1_1\text{-}\mathsf{AC}_0)$.

Proof. For (a): This follows easily after inspecting the definition of $\mathrm{V}^\star_t(x)$. For (b): $(t \in X)^\star \in \Pi^1_0$ follows from (a). Given $\mathbb{P} \in \mathrm{Ty}{\restriction}$, we first note that then by definition, it can only be the case that $\mathbb{P}$ is N, $\overline{\mathsf{N}}$, or $\{x\colon A\}$ for some $A \in \mathrm{For}^+$ such that A does not contain any $\mathsf{I}_{\mathbb{P}',\mathbb{Q}'}$ expression or type variable. By (a) and Definition 7.18, one can easily verify that $(t \in \mathbb{P})^\star \in \Pi^1_0$ holds.

For (c): $\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(\mathcal{B}, r)$ translates to the formula

$$(r \in \mathbb{P})^\star \wedge \forall y((y \in \mathbb{P})^\star \to (\langle y, x\rangle \in \mathbb{Q})^\star(r/x) \to y \in \mathcal{B})$$

and then the claim follows from Proposition 7.16 and (b), using the assumption $B \in \Pi^1_n(\mathsf{T})$ and that $\mathbb{P}, \mathbb{Q} \in \mathrm{Ty}\!\restriction$ holds.

For (d): We have $\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(X, x) \in \Pi^1_0$ by the second claim of (c), and further with (b) and after inspecting ($\star$-$\mathsf{I}_{\mathbb{P},\mathbb{Q}}$) on page 39 and ($\star$-$\mathrm{Cl}_{\mathbb{P},\mathbb{Q}}$) on page 39, the claim becomes clear.

For (e): We prove here a more general statement

$$\left.\begin{array}{rcl} A \in \mathrm{For}^+ & \Longrightarrow & A^\star \in \Pi^1_1(\Sigma^1_1\text{-}\mathsf{AC}_0) \\ \neg A \in \mathrm{For}^+ & \Longrightarrow & \neg A^\star \in \Pi^1_1(\Sigma^1_1\text{-}\mathsf{AC}_0) \end{array}\right\} \qquad (*)$$

and by induction on the build-up of the $\mathcal{L}_{\mathsf{FIT}}$ formula A. Now, let $A \in \mathrm{For}^+$ or $\neg A \in \mathrm{For}^+$ be given. Note that A cannot be of the form $\forall X A_0$ or $\exists X A_0$ because of the definition of For^+.

1. *Base case:* If A is of the form $t \in \mathsf{U}$, $t\!\downarrow$, or $s = t$, we have $A^\star \in \Pi^1_0$ and are done.

2. *Step case $t \in \mathbb{P}$:* If A is $t \in \mathbb{P}$ with $\mathbb{P} \in \mathrm{Ty}$, then $A \in \mathrm{For}^+$ must hold. Because of (b), we also only need to consider the case where $\mathbb{P} \notin \mathrm{Ty}\!\restriction$ and $\mathbb{P}$ is not a type variable. Hence, $\mathbb{P}$ is either of the form $\mathsf{I}_{\mathbb{P}',\mathbb{Q}'}$ with $\mathbb{P}', \mathbb{Q}' \in \mathrm{Ty}\!\restriction$ or $\mathbb{P}$ is of the form $\{z\colon B\}$ for some $B \in \mathrm{For}^+$.

In the first case, we get $A^\star \in \Pi^1_1(\Sigma^1_1\text{-}\mathsf{AC}_0)$ from (d). For the second case, recall that $(t \in \{z\colon B\})^\star$ equals $\exists x(\mathrm{V}^\star_t(x) \wedge B^\star(x/z))$ and note that by the induction hypothesis for $(*)$ with $B(x/z)$, we get $A_0(U, x) \in \Pi^1_0$ for some set variable U such that $B^\star(x/z)$ is equivalent to $\forall X A_0(X, x)$ over $\Sigma^1_1\text{-}\mathsf{AC}_0$. Hence $(t \in \{z\colon B\})^\star$ is equivalent to

$$\exists x \forall X(\mathrm{V}^\star_t(x) \wedge A_0(X, x)) \qquad (26)$$

By letting $A'_0(W) := \exists x\big(\mathrm{V}^\star_t(x) \wedge A_0((W)_x, x)\big)$, we get $\forall X A'_0(X) \in \Pi^1_1$ and it only remains to show that (26) and $\forall X A'_0(X)$ are equivalent over $\Sigma^1_1\text{-}\mathsf{AC}_0$, i.e.,

$$\Sigma^1_1\text{-}\mathsf{AC}_0 \vdash \exists x \forall X\big(\mathrm{V}^\star_t(x) \wedge A_0(X, x)\big) \leftrightarrow \forall X \exists x\big(\mathrm{V}^\star_t(x) \wedge A_0((X)_x, x)\big) \qquad (**)$$

The "$\to$"-direction holds already over ACA_0: In order to show $A_0((X)_y, y)$ for some y for any given set X, take x that is given from the left-hand side of $(**)$. Then use (ACA) to get Z such that $z \in Z \leftrightarrow z \in (X)_x$ holds, then the left-hand side of $(**)$ yields $\mathrm{V}^\star_t(x) \wedge A_0(Z, x)$, i.e., $\mathrm{V}^\star_t(x) \wedge A_0((X)_x, x)$. For the "$\leftarrow$"-direction, we can work with the contraposition of $(**)$ and apply $(\Sigma^1_1\text{-}\mathsf{AC})$.

3. *Otherwise:* The remaining cases follow from standard arguments involving $\Sigma^1_1\text{-}\mathsf{AC}_0$, the induction hypothesis, and Proposition 7.16. We refer to [Ran15] for details.

For (f): We prove this for $\mathbb{F} \in \mathrm{FT}$ with $\mathbb{F} = \mathbb{P}_0 \to \ldots \to \mathbb{P}_n$ by induction on $n \in \mathbb{N}$: If $n = 0$ holds, then we have $\mathbb{F} \in \mathrm{Ty}$ and $(t \in \mathbb{F}) \in \mathrm{For}^+$, so we can use (e). If $n > 0$ holds, then let $\mathbb{F}' := \mathbb{P}_1 \to \ldots \to \mathbb{P}_n$. Now, $(t \in \mathbb{F})^\star$ translates to $\forall x\big((x \in \mathbb{P})^\star \to (tx \in \mathbb{F}')^\star\big)$. By (e) and the induction hypothesis, we get $(x \in \mathbb{P})^\star \in \Pi^1_1(\Sigma^1_1\text{-}\mathsf{AC}_0)$ and $(tx \in \mathbb{F}')^\star \in \Pi^1_2(\Sigma^1_1\text{-}\mathsf{AC}_0)$. By Corollary 7.17, we get $(t \in \mathbb{F})^\star \in \Pi^1_2(\Sigma^1_1\text{-}\mathsf{AC}_0)$. □

Theorem 7.22. Π^1_3-RFN_0 *proves* $A^\star$ *for every instance* A *of* (FT-ID).

Proof. Let A be an instance of (FT-ID), say

$$\mathrm{Cl}_{\mathbb{P},\mathbb{Q}}(\Lambda z.tz \in \mathbb{F}) \to t \in (\mathsf{I}_{\mathbb{P},\mathbb{Q}} \twoheadrightarrow \mathbb{F})$$

with $\mathbb{F} \in \mathrm{FT}$. Similar as in ($\star$-$\mathrm{Cl}_{\mathbb{P},\mathbb{Q}}$) on page 39, we have with $\mathcal{A} := \Lambda z.(tz \in \mathbb{F})^\star$ that $A^\star$ translates to

$$\forall x\big(\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(\mathcal{A}, x) \to \mathcal{A}(x)\big) \to \forall x\big((x \in \mathsf{I}_{\mathbb{P},\mathbb{Q}})^\star \to \mathcal{A}(x)\big)$$

and therefore we assume (with a slight renaming of bound variables to make the following more readable) that

$$\forall y\big(\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(\mathcal{A}, y) \to \mathcal{A}(y)\big) \tag{27}$$

holds. Due to Lemma 7.21.(f), we know that a formula $B \in \Pi^1_2$ exists such that

$$\Sigma^1_1\text{-}\mathsf{AC}_0 \vdash B \leftrightarrow \mathcal{A}(y)$$

holds. For $\mathcal{B} := \Lambda y.B$, we get from Corollary 7.17 and Lemma 7.21.(c) a formula $C \in \Pi^1_3$ such that

$$\mathsf{ACA}_0 \vdash C \leftrightarrow \forall y(\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(\mathcal{B}, y) \to \mathcal{B}(y)) \tag{28}$$

holds. Note that this holds over ACA_0 since we work with $B \in \Pi^1_2$ instead of $(ty \in \mathbb{F})^\star$. Moreover, we have over Σ^1_1-AC_0 that (27) is equivalent to $\forall y(\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(\mathcal{B}, y) \to \mathcal{B}(y))$ and we proceed now by assuming that the conclusion in $A^\star$ is false and will derive a contradiction from this. So, let a_0 be such that

$$(x \in \mathsf{I}_{\mathbb{P},\mathbb{Q}})^\star(a_0/x) \wedge \neg\mathcal{A}(a_0) \tag{29}$$

holds and note that the formula $\neg\mathcal{A}(a_0)$ (which is $\neg(tz \in \mathbb{F})^\star(a_0/z)$) is equivalent over Σ^1_1-AC_0 to $\neg B(a_0/y)$. Note that $\neg B(a_0/y)$ is equivalent to a Π^1_3 formula, and since we have $C \in \Pi^1_3$, there exists by Proposition 7.16 some $D \in \Pi^1_3$ that is provably equivalent over ACA_0 to $C \wedge \neg B(a_0/y)$. Then due to Corollary 7.11, we can work with Π^1_3-RFN_0 to apply (Π^1_3-RFN) to D and thus obtain an ω-model M of ACA_0 such that the following holds:

$$\forall y(\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(\mathcal{B}, y)^M \to \mathcal{B}(y)^M) \tag{30}$$

$$\neg B^M(a_0/y) \tag{31}$$

Relativization to M in (30) holds essentially because of the equivalence in (28) being provable over ACA_0. Now, (30) unfolds by Definition 7.1 and the build-up of $\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(\mathcal{B}, y)$ to

$$\forall y(\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(\Lambda y.B^M, y) \to B^M) \tag{32}$$

Since B^M is arithmetical, (ACA) provides a set X_0 such that we have

$$\forall y(y \in X_0 \leftrightarrow B^M) \tag{33}$$

$$\forall y(\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(X_0, y) \to y \in X_0) \tag{34}$$

Now, after recalling ($\star$-$\mathsf{I}_{\mathbb{P},\mathbb{Q}}$) on page 39, we instantiate $(x \in \mathsf{I}_{\mathbb{P},\mathbb{Q}})^\star(a_0/x)$ from (29) with X_0 and (34). We obtain then $(x \in X_0)^\star(a_0/x)$, i.e., $\exists z(\mathrm{V}^\star_x(z) \wedge z \in X_0)(a_0/x)$ which is equivalent to $a_0 \in X_0$ since $\mathrm{V}^\star_x(z)$ is just $x = z$. But then we get $B^M(a_0/y)$ by (33) which is a contradiction to (31) and we have proven the lemma. □

Remark 7.23. In the previous proof, we considered (32) as the pivotal property for the used proof method because it allowed us to internalize an argument withing the ω-model M. In particular, we needed that the positive operator form $\mathrm{Acc}^{\star}_{\mathbb{P},\mathbb{Q}}$ has the property described by Lemma 7.21.(c) with T being ACA_0. A conceptually similar proof in the setting of Π^1_2-RFN_0 and using similar standard results from the area of subsystems of second order arithmetic can be found in [AR10], treating the embedding of the theory ID^*_1 of positive induction into Π^1_2-RFN_0.

Now, turning to the question if our proof method would also work for arbitrary positive operator forms $\mathfrak{A}$, we point out that a direct embedding of TID into Π^1_3-RFN_0 can be carried out almost literally as the embedding of FIT into Π^1_3-RFN_0. More precisely, the previous lemmas can be reformulated in a very similar way so that they work for TID as well. The pivotal property to make the proof work would again correspond to (32), and essentially because Acc in the setting of TID has a similar bounded complexity as $\mathrm{Acc}^{\star}_{\mathbb{P},\mathbb{Q}}$ here. The latter means that for (28) in the proof of Theorem 7.22, we used that we had the property $\mathrm{Acc}^{\star}_{\mathbb{P},\mathbb{Q}}(\Lambda y.B, x) \in \Pi^1_2(\mathsf{ACA}_0)$ at hand for $B \in \Pi^1_2$, namely as provided by Lemma 7.21.(c).

Continuing from the perspective of TID, we shall consider for a moment its natural generalization $\mathsf{TID}^{\mathbf{f}}$ (where $\mathbf{f}$ stands for *full*) that allows for arbitrary arithmetical operator forms $\mathfrak{A}$. So, having $\mathfrak{A}$ instead of $\mathrm{Acc}^{\star}_{\mathbb{P},\mathbb{Q}}$ or Acc in (28), it would not always be possible to obtain a property such as $\mathfrak{A}(\mathcal{B}, x) \in \Pi^1_2(\mathsf{ACA}_0)$, nor can we expect that $G' \in \Pi^1_2$ exists that is equivalent over Σ^1_1-AC_0 or Π^1_3-RFN_0 to $\mathfrak{A}(\mathcal{B}, x)$. Comparing this with the mentioned embedding of ID^*_1 into Π^1_2-RFN_0 from [AR10], we note that essentially only Π^1_1 formulas B needed to be considered there, and since a formula such as $\mathfrak{A}(\Lambda z.B, t)$ can be proven to be equivalent over Σ^1_1-AC_0 to a Π^1_1 formula G', one can continue the proof with this G'.

For an embedding of $\mathsf{TID}^{\mathbf{f}}$ into Π^1_3-RFN_0 where we cannot control anymore the syntactical complexity of the positive operator forms $\mathfrak{A}$, we apparently cannot directly apply the method of this section. As we shall describe in the conclusion of this article (see Section 8), we remark here that the desired upper bound for $\mathsf{TID}^{\mathbf{f}}$ can be obtained by turning to the setting of set-theory.

Theorem 7.24. *Over* Π^1_3-RFN_0*, the following holds.*

(a) $A^\star$ *holds for every formula* A *from axiom group* ***I.*** *of* FIT.

(b) $A^\star$ *holds for every instance* A *of the* N*-induction scheme* (FT-Ind) *of* FIT.

(c) $A^\star$ *holds for every instance* A *of the comprehension scheme* (CA^+) *of* FIT.

(d) $A^\star$ *holds for every instance* A *of the closure axiom* (FT-Cl) *of* FIT.

Proof. For (a): Note that according to Definition 7.18, the type N has no special role in the translation $A^\star$ of any of the formulas A given in the axiom group **I.** of FIT. As mentioned in Definition 7.18, we assume a standard interpretation of the constants k and s with the properties that we need for such a translation to be adequate. It is well-known that the combinators are available as partial recursive functions in the sense given here. Moreover, it is also more or less obvious that the interpretation of the remaining constants has the properties needed to make the translation of the remaining formulas in axiom group **I.** go through.

For (b): Over ACA_0, we have that (Π^1_3-RFN) implies transfinite induction for Π^1_2 formulas, and thus complete induction along the natural numbers for Π^1_2 formulas. For this, see in [Sim09] Theorem VIII.5.12 and in particular Exercise VIII.5.15, while noting there that

Σ^1_4-RFN_0 is equivalent to Π^1_3-RFN_0. Now, let A be an instance $t0 \in \mathbb{F} \wedge (\forall x \in \mathsf{N})(tx \in \mathbb{F} \to tx' \in \mathbb{F}) \to t \in (\mathsf{N} \twoheadrightarrow \mathbb{F})$ of the N-induction scheme (FT-Ind) of FIT, where $\mathbb{F} \in \mathrm{FT}$ holds. By setting $\mathcal{B} := \Lambda z.\big(tz \in \mathbb{F}\big)^\star$, we have that $A^\star$ is equivalent over ACA_0 to

$$\mathcal{B}(0) \wedge \forall x(\mathcal{B}(x) \to \mathcal{B}(x+1)) \to \forall x(\exists y(\mathrm{V}^\star_x(y)) \to \mathcal{B}(x)) \tag{35}$$

since $\mathcal{B}(x+1)$ is equivalent to $\big(t(\mathsf{s_N} x) \in \mathbb{F}\big)^\star$. For the latter, note that $\big(t(\mathsf{s_N} x) \in \mathbb{F}\big)^\star$ is $\exists y(\mathrm{V}^\star_{t(\mathsf{s_N} x)}(y) \wedge (y \in \mathbb{F})^\star)$ and that this is equivalent to

$$\exists y, z_1, z_2(\mathrm{V}^\star_t(z_1) \wedge \{\mathsf{s}^\star_\mathsf{N}\}(x) \simeq z_2 \wedge \{z_1\}(z_2) \simeq y \wedge (y \in \mathbb{F})^\star)$$

which again simplifies to

$$\exists y, z_1(\mathrm{V}^\star_t(z_1) \wedge \{z_1\}(x+1) \simeq y \wedge (y \in \mathbb{F})^\star)$$

and this is equivalent to $\mathcal{B}(x+1)$. Now arguing over Π^1_3-RFN_0, we have that (35) is equivalent to an instance of complete induction along the natural numbers for a Π^1_2 formula (use Lemma 7.21.(f)) and hence we are done.

For (c): Let A be an instance of (CA^+), say $y \in \{x\colon B\} \leftrightarrow B(y/x)$ with $B \in \mathrm{For}^+$. Then, $A^\star$ yields

$$\exists x(\mathrm{V}^\star_y(x) \wedge B^\star) \leftrightarrow (B(y/x))^\star$$

which is equivalent to $(B(y/x))^\star \leftrightarrow (B(y/x))^\star$ and hence a tautology.

For (d): Let $A := \mathrm{Cl}_{\mathbb{P},\mathbb{Q}}(\Lambda z.z \in \mathsf{I}_{\mathbb{P},\mathbb{Q}})$ be an instance of (FT-Cl). Following ($\star$-$\mathrm{Cl}_{\mathbb{P},\mathbb{Q}}$) on page 39 and in order to show $A^\star$, assume for $\mathcal{A} := \Lambda z.(z \in \mathsf{I}_{\mathbb{P},\mathbb{Q}})^\star$ that we have $\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(\mathcal{A}, z_0)$ for some z_0, and we aim to prove $\mathcal{A}(z_0)$, i.e.,

$$\big(\forall X(\forall x(\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(X,x) \to x \in X) \to (z \in X)^\star)\big)(z_0/z)$$

and in order to prove this, let X_0 be given such that

$$\forall x(\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(X_0,x) \to x \in X_0) \tag{36}$$

holds and show $z_0 \in X_0$. We have $\forall z((z \in \mathsf{I}_{\mathbb{P},\mathbb{Q}})^\star \to z \in X_0)$ due to (36) and the definition of $(z \in \mathsf{I}_{\mathbb{P},\mathbb{Q}})^\star$, i.e., we have $\forall z(\mathcal{A}(z) \to z \in X_0)$. So, the latter yields

$$\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(\mathcal{A}, z_0) \to \mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(X_0, z_0)$$

because $\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}$ is a *positive* operator form. We hence get $\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(X_0, z_0)$ from our assumption $\mathrm{Acc}^\star_{\mathbb{P},\mathbb{Q}}(\mathcal{A}, z_0)$, and with (36) we are done. □

Corollary 7.25 (Embedding FIT into Π^1_3-RFN_0)**.** *Let $A \in \mathcal{L}_\mathsf{FIT}$. Then we have*

$$\mathsf{FIT} \vdash A \implies \Pi^1_3\text{-}\mathsf{RFN}_0 \vdash A^\star$$

Proof. Let A be any $\mathcal{L}_\mathsf{FIT}$ formula. Due to Theorems 7.22 and 7.24, it remains only to show that the logical part of FIT embeds into Π^1_3-RFN_0 in the following sense:

$$\mathsf{LPT} \vdash A \implies \Pi^1_3\text{-}\mathsf{RFN}_0 \vdash A^\star$$

Assume $\mathsf{LPT} \vdash A$ with respect to any sound Hilbert calculus that may have been chosen in Definition 3.11. We prove $\Pi^1_3\text{-}\mathsf{RFN}_0 \vdash A^\star$ by induction on the definition of the derivability notion $\mathsf{LPT} \vdash A$. It is clear from Definition 7.18 that the translation of the propositional axioms and rules are derivable in the setting of $\mathcal{L}^2_{\mathsf{PA}}$. Similarly, the equality axioms and the translation of the quantificational axioms and rules for individual variables is stable, while the definedness axioms become trivial.

Now, we consider the remaining quantificational axioms and rules for type variables, we have the following cases (and given $\mathcal{L}_{\mathsf{FIT}}$ formulas A, B):

1. For axiom $A := \forall X B \to B(\mathbb{P}/X)$: Then $A^\star$ is

$$\forall z(B^\star\big((\Lambda a.\Pi^1_1(z,a))/X\big)) \to B^\star\big(\Lambda a.(a \in \mathbb{P})^\star/X\big)$$

and the claim follows due to $(a \in \mathbb{P})^\star \in \Pi^1_1(\Sigma^1_1\text{-}\mathsf{AC}_0)$ from Lemma 7.21.(e) and since $\Pi^1_1(x,y)$ denotes a universal Π^1_1 formula.

2. For axiom $A := B(\mathbb{P}/X) \to \exists X B$: Use the contraposition of A and argue as in the previous case.

3. For the logical rule

$$\frac{A \to B}{A \to \forall X B}$$

with X not occurring free in A, we get $\Pi^1_3\text{-}\mathsf{RFN}_0 \vdash A^\star \to B^\star$ form the induction hypothesis. Since the underlying calculus for $\Pi^1_3\text{-}\mathsf{RFN}_0$ from Definition 4.9 is closed under substitution and X does not occurring in A, we obtain $\Pi^1_3\text{-}\mathsf{RFN}_0 \vdash A^\star \to B^\star\big((\Lambda a.\Pi^1_1(z,a))/X\big)$. If we further let $z \in \mathrm{FV}(A^\star)$, we eventually get $A^\star \to \forall z(B^\star\big((\Lambda a.\Pi^1_1(z,a))/X\big))$.

4. For the remaining logical rule of the existential quantifier, this is holds similarly. □

8. Conclusion and Further Remarks

Recapitulating the results of this article, we obtained theories FIT and TID that both have the small Veblen ordinal $\vartheta\Omega^\omega$ as their proof-theoretic ordinal, while FIT is a natural extension of Feferman's two-sorted theory $\mathrm{QL}(\mathsf{F_0}\text{-}\mathsf{IR}_N)$ from [Fef92] and TID becomes from this perspective a natural subsystem of ID_1. Moreover, we used techniques from the realm of predicative proof-theory in order to obtain a wellordering proof for TID (and hence for FIT).

With regard to the upper-bound results, we embedded FIT into the subsystem $\Pi^1_3\text{-}\mathsf{RFN}_0$ of second order arithmetic, while exploiting the Π^1_1 definability of a least fixed point in such a setting. This method can be used almost literally for embedding TID directly into $\Pi^1_3\text{-}\mathsf{RFN}_0$ (see [Ran15] for this). This approach seems to fail though if we extend TID to a theory $\mathsf{TID}^{\mathbf{f}}$ for general typed inductive definitions with the *full* range of positive arithmetical operator forms (as described in Remark 7.23). A way to avoid this obstacle is to shift the setting to set-theory rather than subsystems of second order arithmetic, namely by exploiting the Σ_1 definability of a least fixed point. Working then in $\mathsf{KP}\omega^- + \Pi_2\text{-}\mathsf{Found}$ from [Rat92] (i.e., Kripke-Platek set-theory with a restricted axiom scheme for foundation) shall suffice to get an analog result as for FIT which we can apply to the theory $\mathsf{TID}^{\mathbf{f}}$. Summing up, what we gain from these embeddings is that extending TID to the theory $\mathsf{TID}^{\mathbf{f}}$ retains the proof-theoretic upper bound $\vartheta\Omega^\omega$ because $\mathsf{KP}\omega^- + \Pi_2\text{-}\mathsf{Found}$ has by [Rat92] the same proof-theoretic strength as $\Pi^1_2\text{-}\mathsf{BI}_0$. Since TID trivially embeds into $\mathsf{TID}^{\mathbf{f}}$ and as depicted in Figure 2, we get that $\mathsf{TID}^{\mathbf{f}}$ has the same proof-theoretic strength as TID (in a similar way as ID_1 and its restriction $\mathsf{ID}_1(\mathrm{Acc})$ to accessible part arithmetical operator forms correspond, see also [BFPS81]).

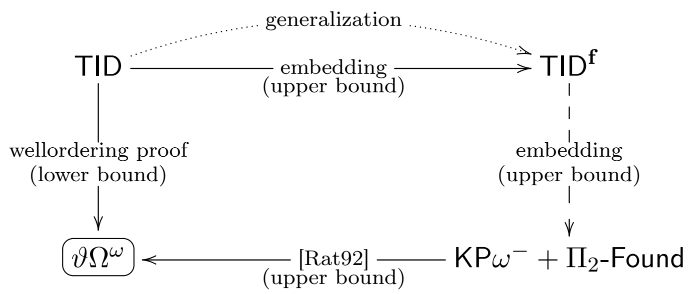

Figure 2: Generalization of TID to the full theory $\mathsf{TID}^{\mathbf{f}}$

We finish with some remarks and conjectures on how to extend the proof-methods from Sections 6 and 7 in order to analyze stronger systems: First, the collections of formulas Pos_0 and Pos_1 suggest already a generalization to collections Pos_n for any $n \geq 2$ in the sense that the correspondence of Pos_1 to *function* types of the form $\mathbb{P}_1 \twoheadrightarrow \ldots \twoheadrightarrow \mathbb{P}_k$ for each $k \in \mathbb{N}$ (i.e., "*level-1-functional* types") passes over to Pos_2 being in correspondence to *level-2-functional* types $\mathbb{F}_1 \twoheadrightarrow \ldots \twoheadrightarrow \mathbb{F}_k$ for each $k \in \mathbb{N}$, and similar for $n > 2$. Such theories are essentially considered in [Ran15] and named TID_n. Moreover, intermediate systems TID_n^+ are considered that are located between TID_n and TID_{n+1}, and which arise essentially by adding a rule of inference (TID^+) of the form

$$(\mathsf{TID}^+)\ \frac{P_{\lhd} t}{\mathrm{Prog}_{\lhd}(\mathcal{B}) \to \mathcal{B}(t)}$$

for *arbitrary* $\mathcal{B} \in \mathcal{L}_{\mathsf{TID}}$ and for each binary relation symbol $\lhd \in \mathcal{L}_{\mathsf{PA}}$ and each term t. As shown in [Ran15],TID_1^+ has the strength of the *large Veblen ordinal* $\vartheta\Omega^\Omega$ and (TID^+) corresponds to a variant of a *bar rule*. This and the results of [Pro16] suggest that by adding the usual bar rule (BR) to the theory Π^1_3-RFN_0 with strength of the small Veblen ordinal $\vartheta\Omega^\omega$, we obtain a theory Π^1_3-RFN_0 + (BR) with strength of the large Veblen ordinal $\vartheta\Omega^\Omega$.

Concerning the wellordering proof for TID_1^+ in [Ran15], we remark here that it is based on the methods from Section 6 by using *Klammersymbols* instead of finitary Veblen functions and by *internalizing* the arity of the finitary Veblen functions within the theory. More precisely and as an informal and intuitive explanation why Corollary 6.17 is the best we can expect from TID, we mention that the method used in the proof of Theorem 6.14 relied on an *external* representation of the finite list of arguments that the finitary Veblen function is applied to. In partiular, induction in the meta-theory has been applied to cope with arbitrary but finite lists of arguments. The proof of Theorem 6.14 is designed for the theory TID, and in order to use it to get beyond the small Veblen ordinal, for instance by working with infinitary Veblen functions or Klammersymbols, the first step is to *internalize the proof* and deal with non-standard argument positions (for which we do not have a denotation in the meta-theory).

For more details and further remarks on the other theories, we refer to [Ran15]. The use of a higher-type functional for iterating the fixed-point construction on Klammersymbols allows to extend the ordinal notation system. Endowed with stronger induction principles that become available in the theories TID_n and TID_n^+, this may lead towards higher ordinals via wellordering proofs based on the accessible part of the primitive recursive ordering of the new ordinal notation system.

A. Appendix: The Ordinal Notation System $(\mathrm{OT}, \prec)$

A.1. The Definition of $(\mathrm{OT}, \prec)$

For carrying out the wellordering proofs in TID, we fix a primitive recursive notation system $(\mathrm{OT}, \prec)$ for ordinals below the small Veblen ordinal. It is based on Lemma 5.17 (i.e., essentially on (7.1)–(7.4) in [Sch54]). Furthermore, [Buc05] inspired the representation of the following material. We point out that the following definition and properties of $(\mathrm{OT}, \prec)$ can be formalized and established within PA and hence within TID.

Definition A.1. Based on the coding machinery mentioned in Definition 4.5, we introduce the following abbreviations:

$$\phi\bar{a}^{(n+1)} := \phi a_1 \dots a_{n+1} := \phi(a_1, \dots, a_{n+1}) := \langle 1, a_1, \dots, a_{n+1} \rangle$$

$$\tilde{1} := \phi 0 \qquad a \oplus b := \begin{cases} a & \text{if } b = 0 \\ \langle 2, a, b \rangle & \text{otherwise} \end{cases}$$

$$\begin{aligned} \mathrm{PT}_+ &:= \{\phi\bar{a}^{(n+1)} \colon a_1 \neq 0 \ \ \& \ \ a_1, \dots, a_{n+1} \in \mathbb{N}\} \\ &= \{a \colon \mathsf{lh}(a) \geq 2 \wedge (a)_0 = 1 \wedge (a)_1 \neq 0\} \\ \mathrm{PT} &:= \mathrm{PT}_+ \cup \{\tilde{1}\} \end{aligned}$$

$$\mathsf{hd}(a) := \begin{cases} a & \text{if } a \in \mathrm{PT} \\ (a)_1 & \text{otherwise} \end{cases} \qquad \mathsf{tl}(a) := \begin{cases} 0 & \text{if } a \in \mathrm{PT} \\ (a)_2 & \text{otherwise} \end{cases}$$

Definition A.2. Moreover, for any binary relation $\lhd$ on $\mathbb{N}$, we define the *(length-sensitive) lexicographic order* $\lhd_{\mathbf{lex}}$ *with respect to* $\lhd$ recursively as follows. $a \lhd_{\mathbf{lex}} b$ holds for any $a, b \in \mathbb{N}$ if and only if:

1. $\mathsf{lh}(a) < \mathsf{lh}(b)$ holds, or
2. $\mathsf{lh}(a) = \mathsf{lh}(b)$ holds and there is some $k < \mathsf{lh}(a)$ with $(a)_k \lhd (b)_k$ such that $(a)_i = (b)_i$ holds for all $i < k$.

Example A.3. Note that $\langle 1, 2 \rangle <_{\mathbf{lex}} \langle 1, 1, 3 \rangle$ holds but not $\langle 1, 1, 3 \rangle <_{\mathbf{lex}} \langle 1, 2 \rangle$ and that $\langle 1, 2 \rangle$ corresponds to $\langle 0, 1, 2 \rangle$ here. If we have $a < b$, then $\langle a, a \rangle <_{\mathbf{lex}} \langle a, b \rangle$ holds but not $\langle b, a \rangle <_{\mathbf{lex}} \langle a, b \rangle$. Note that $\lhd_{\mathbf{lex}}$ is primitive recursive if $\lhd$ is.

Definition A.4. Motivated by Corollary 5.14 and Lemma 5.17, we now define *simultaneously* the primitive recursive set OT of ordinal notations and the binary primitive recursive relation $\prec$ on OT. We have $c \in \mathrm{OT}$ if and only if one of the following cases holds:

1. $c = 0$ or $c = \tilde{1}$ holds.
2. $c \in \mathrm{PT}_+$ holds with $c = \phi\bar{a}^{(m+1)}\bar{0}^{(k)}$ for some $a_1, \dots, a_{m+1} \in \mathrm{OT}$ such that $a_{m+1} \neq 0$ and one of the following cases holds:
   (i) $a_{m+1} \notin \mathrm{PT}_+$,
   (ii) $a_{m+1} \in \mathrm{PT}_+$ and $a_{m+1} \prec_{\mathbf{lex}} c$, or
   (iii) $a_{m+1} \in \mathrm{PT}_+$, $c \prec_{\mathbf{lex}} a_{m+1}$ and $a_{m+1} \preceq a_j$ holds for some $1 \leq j \leq m$.
3. $c = a \oplus b$ holds for some $a, b \in \mathrm{OT}$ and such that $a \in \mathrm{PT}$, $b \neq 0$, and $\mathsf{hd}(b) \preceq a$ hold.

With $a \preceq b$, we abbreviate in general $a \prec b \vee (a = b \wedge a \in \mathrm{OT} \wedge b \in \mathrm{OT})$. Now, $a \prec b$ holds if and only if $a, b \in \mathrm{OT}$ and one of the following cases hold:

1. $a = 0$ and $b \neq 0$ hold.
2. $a = \tilde{1}$, $b \neq 0$, and $b \neq \tilde{1}$ hold.
3. $a \in \mathrm{PT}_+$ and $b \in \mathrm{PT}_+$ hold with $a = \phi\bar{a}^{(m+1)}\bar{0}^{(k)}$ and $b = \phi\bar{b}^{(n+1)}\bar{0}^{(l)}$ such that $a_{m+1}, b_{n+1} \neq 0$ and one of the following cases hold:
   (i) $a \prec_{\mathbf{lex}} b$ and $a_i \prec b$ for all $1 \leq i \leq m+1$, or
   (ii) $b \prec_{\mathbf{lex}} a$ and $a \prec b_{n+1}$ or $a \preceq b_j$ for some $1 \leq j \leq n$.
4. $a = a_1 \oplus a_2$, $b = b_1 \oplus b_2$, and $a_1, b_1 \in \mathrm{PT}$ hold with $a_2 \neq 0$ or $b_2 \neq 0$ such that one of the following cases holds:
   (i) $a_1 \prec b_1$ or
   (ii) $a_1 = b_1$ and $a_2 \prec b_2$.

We use common abbreviations in combination with these notions, e.g., $a \not\prec b$ abbreviates $\neg(a \prec b)$, $(\forall x \preceq t)A$ abbreviates $\forall x(x \preceq t \to A)$, and analogously $(\exists x \preceq t)A$ abbreviates $\exists x(x \preceq t \wedge A)$.

Remark A.5. Let $a = \phi\bar{a}^{(m+1)}\bar{0}^{(k)}$ and $b = b_1 \oplus b_2$ with $b_2 \neq 0$, then we obviously have $a_1, \ldots, a_{m+1} <_{\mathbb{N}} a$ and $b_1, b_2 <_{\mathbb{N}} b$. Moreover $a \neq 0$ and $b \neq 0$ hold.

Theorem A.6. $(\mathrm{OT}, \prec)$ *and* $(\mathrm{PT}_{\mathrm{OT}}, \prec_{\mathbf{lex}})$ *are strict total orders, where we let here* $\mathrm{PT}_{\mathrm{OT}} := \{\phi\bar{a}^{(n+1)} \in \mathrm{PT} \colon a_1, \ldots, a_{n+1} \in \mathrm{OT}\}$.

Proof. By a straightforward but long and cumbersome induction on the build-up of OT, see [Ran15] for details. □

Lemma A.7. *Let* $a \in \mathrm{OT}$.

(a) If $a = a_1 \oplus a_2$ *with* $a_1 \in \mathrm{PT}$ *and* $a_2 \neq 0$*, then* $a_1, a_2 \prec a$.

(b) If $a = \phi\bar{a}^{(m+1)}\bar{0}^{(k)}$ *with* $a_{m+1} \neq 0$*, then* $a_i \prec a$ *for each* $1 \leq i \leq m+1$.

Proof. See [Ran15]. □

A.2. Ordinal Arithmetic within $(\mathrm{OT}, \prec)$

We point out that the following definitions and properties can be formalized and established within PA and hence within TID.

Definition A.8. In order to simulate ordinal addition and the finitary Veblen functions within OT, we introduce the following primitive recursive functions on natural numbers.

(a) For each $a, b \in \mathbb{N}$, we define

$$a \tilde{+} b := \begin{cases} a & \text{if } a \in \mathrm{OT} \text{ and } b = 0 \\ b & \text{if } a = 0 \text{ and } b \in \mathrm{OT} \setminus \{0\} \\ \mathsf{hd}(a) \oplus (\mathsf{tl}(a) \tilde{+} b) & \text{if } a, b \in \mathrm{OT} \setminus \{0\} \text{ and } \mathsf{hd}(b) \preceq \mathsf{hd}(a) \\ b & \text{if } a, b \in \mathrm{OT} \setminus \{0\} \text{ and } \mathsf{hd}(a) \prec \mathsf{hd}(b) \\ 0 \oplus \tilde{1} & \text{otherwise, i.e., if } a \notin \mathrm{OT} \text{ or } b \notin \mathrm{OT} \end{cases}$$

(b) For each $n \in \mathbb{N}$ and $\bar{a}^{(n+1)} \in \mathbb{N}$, we define:

$$\tilde{\varphi}_{n+1}(\bar{a}^{(n+1)}) := \begin{cases} \phi\bar{a}^{(n+1)} & \text{if } \phi\bar{a}^{(n+1)} \in \mathrm{OT} \\ \mathrm{cr}(\langle\bar{a}^{(n+1)}\rangle) & \text{if } \phi\bar{a}^{(n+1)} \notin \mathrm{OT},\ a_1,\dots,a_{n+1} \in \mathrm{OT}, \text{ and } a_1 \neq 0 \\ \tilde{\varphi}_n(a_2,\dots,a_{n+1}) & \text{if } \phi\bar{a}^{(n+1)} \notin \mathrm{OT},\ a_1,\dots,a_{n+1} \in \mathrm{OT}, \text{ and } a_1 = 0 \\ 0 \oplus \tilde{1} & \text{otherwise, i.e., if } a_j \notin \mathrm{OT} \text{ for some } 1 \leq j \leq n+1 \end{cases}$$

and

$$\mathrm{cr}(\langle\bar{a}^{(n)}\rangle) := \begin{cases} 0 & \text{if } n = 0 \\ \mathrm{cr}(\langle\bar{a}^{(n-1)}\rangle) & \text{if } n \neq 0 \text{ and } a_n = 0 \\ a_n & \text{otherwise} \end{cases}$$

and since the index $n+1$ will be clear from the context, we also just write $\tilde{\varphi}(a_1,\dots,a_{n+1})$ in order to denote $\tilde{\varphi}_{n+1}(a_1,\dots,a_{n+1})$.

Remark A.9. Note that $n \neq 0$ holds in the third clause of the definition of $\tilde{\varphi}_{n+1}(\bar{a}^{(n+1)})$. Furthermore, the naming of $\mathrm{cr}\colon \mathbb{N} \to \mathbb{N}$ is motivated from the intention of returning a fixed-point of $\tilde{\varphi}_{n+1}$ and that fixed-points $\beta = \varphi(\alpha,\beta)$ of the binary Veblen function are sometimes called *critical* in the literature.

Definition A.10. We further introduce the following notations for every $a, x \in \mathbb{N}$:

$$a \mathbin{\tilde{\cdot}} x := \begin{cases} 0 & \text{if } x = 0 \\ a \oplus (a \mathbin{\tilde{\cdot}} x_0) & \text{if } x = x_0 +_{\mathbb{N}} 1 \end{cases}$$

$$\tilde{\varepsilon}_a := \tilde{\varphi}(\tilde{1}, a)$$

$$\tilde{\omega}^a := \tilde{\varphi}(a)$$

$$\tilde{\omega}_x(a) := \begin{cases} a & \text{if } x = 0 \\ \tilde{\omega}^{\tilde{\omega}_{x_0}(a)} & \text{if } x = x_0 +_{\mathbb{N}} 1 \end{cases}$$

$$\tilde{\omega} := \tilde{\omega}^{\tilde{1}} (= \phi(\phi 0))$$

Definition A.11.

$$\mathrm{last}(a) := \begin{cases} \mathrm{last}(a_2) & \text{if } a = a_1 \oplus a_2 \text{ and } a_2 \neq 0 \\ a & \text{otherwise} \end{cases}$$

$$\mathrm{Lim} := \{a \in \mathrm{OT}\colon a \neq 0 \wedge \mathrm{last}(a) \neq \tilde{1}\} \qquad \mathrm{Suc} := \{a \in \mathrm{OT}\colon \mathrm{last}(a) = \tilde{1}\}$$

Elements of Lim are called *limits* and elements of Suc are called *successors*.

Lemma A.12. *For every $a, b, a_1, \dots, a_{n+1}, c \in \mathrm{OT}$ and $n, m \in \mathbb{N}$, we have the following:*

(a) $(a \preceq c$ *and* $c \prec a \tilde{+} b) \implies c = a \tilde{+} d$ *for some* $d \in \mathrm{OT}$ *with* $d \prec b$.

(b) $\tilde{\varphi}(\bar{x}^{(n+1)}) \in \mathrm{OT} \iff x_1, \dots, x_{n+1} \in \mathrm{OT}$ *for each* $x_1, \dots, x_{n+1} \in \mathbb{N}$.

(c) $x \tilde{+} y \in \mathrm{OT} \iff x, y \in \mathrm{OT}$ *for each* $x, y \in \mathbb{N}$.

(d) $(a_1 \neq 0 \ \& \ a_{m+1} \notin \mathrm{PT}_+ \cup \{0\}) \implies \tilde{\varphi}(\bar{a}^{(m+1)}, \bar{0}^{(k)}) = \phi\bar{a}^{(m+1)}\bar{0}^{(k)}$.

Proof. Mostly by induction on the build-up of the involved objects in OT, see [Ran15] for details. □

A.3. Semantics of $(\mathrm{OT}, \prec)$

Definition A.13.

(a) $f_\omega(\gamma) := \omega^\gamma$ for all $\gamma \in \mathrm{On}$.

(b) $\nu_n := f_\omega\binom{1}{n}$ for each $n < \omega$.

Lemma A.14.

(a) $\sup_{n<\omega} \nu_n = \vartheta\Omega^\omega$.

(b) $\nu_n < \nu_{n+1}$ *for all* $n \in \mathbb{N}$.

(c) $\gamma < \vartheta\Omega^\omega \implies \nu_n \leq \gamma < \nu_{n+1}$ *for some* $n \in \mathbb{N}$.

Proof. See [Ran15]. □

Definition A.15. We define $\mathrm{o}(a) \in \mathrm{On}$ and $|a|_\prec \in \mathrm{On}$ for each $a \in \mathrm{OT}$ recursively as follows:

$$\mathrm{o}(a) := \begin{cases} 0 & \text{if } a = 0 \\ \mathrm{o}(a_1) + \mathrm{o}(a_2) & \text{if } a = a_1 \oplus a_2 \text{ with } a_2 \neq 0 \\ \varphi(\mathrm{o}(a_1), \ldots, \mathrm{o}(a_n)) & \text{if } a = \phi a_1 \ldots a_n \text{ with } n \geq 1 \end{cases}$$
$$|a|_\prec := \sup\{|b|_\prec + 1 \colon b \prec a\}$$

where 1 denotes the first non-zero ordinal in the definition of $|a|_\prec$.

Theorem A.16.

(a) With $a \mapsto \mathrm{o}(a)$*, we have an order isomorphism between* $(\mathrm{OT}, \prec)$ *and* $(\vartheta\Omega^\omega, <)$.

(b) $|a|_\prec = \mathrm{o}(a)$ *for each* $a \in \mathrm{OT}$.

Proof. See [Ran15]. □

A.4. Fundamental Sequences within $(\mathrm{OT}, \prec)$

Note that the results from Subsections A.1 and A.2 are provable within PA and hence within TID. With regard to the wellordering proofs in Section 6, we shall render this fact more precise for some properties of fundamental sequences that we are now going to introduce.

Definition A.17. *Fundamental sequences* for limit notations $d \in \mathrm{Lim}$ are defined within PA by means of a binary primitive recursive function L whose defining equations are described as follows, where a, x range over natural numbers and where we write $a[x]$ in order to denote $L(a, x)$.

- If $d = 0$ or $d \notin \mathrm{OT}$, then

$$d[x] := 0$$

- If $d \in \mathrm{Suc}$ with $d = d_0 \tilde{+} \tilde{1}$, then

$$d[x] := d_0$$

- If $d \in \mathrm{Lim}$ and $d = a \oplus b$ with $a \in \mathrm{OT}$ and $b \in \mathrm{Lim}$, then
$$d[x] := a \mathbin{\tilde{+}} b[x]$$
- If $d \in \mathrm{Lim}$ and $d = \phi a$ with $a \neq 0$, then
$$d[x] := \begin{cases} \tilde{\omega}^{a_0} \mathbin{\tilde{\cdot}} (x +_{\mathbb{N}} 1) & \text{if } a = a_0 \mathbin{\tilde{+}} \tilde{1} \\ \tilde{\omega}^{a[x]} & \text{otherwise} \end{cases}$$
- If $d \in \mathrm{Lim}$ with $d = \phi \bar{a}^{(m)} b \bar{0}^{(k)} c$ for some $\bar{a}^{(m)}, b, c \in \mathrm{OT}$ with $b \neq 0$ and $m, k \in \mathbb{N}$, then
$$d[0] := \begin{cases} \tilde{\varphi}(\bar{a}^{(m)}, b, \bar{0}^{(k)}, c[0]) & \text{if } c \in \mathrm{Lim} \\ \tilde{\varphi}(\bar{a}^{(m)}, b[0], \bar{0}^{(k+1)}) & \text{if } c = 0 \text{ and } b \in \mathrm{Lim} \\ \tilde{1} & \text{if } c = 0 \text{ and } b \in \mathrm{Suc} \\ \tilde{\varphi}(\bar{a}^{(m)}, b, \bar{0}^{(k)}, c[0]) \mathbin{\tilde{+}} \tilde{1} & \text{otherwise, i.e., if } c \in \mathrm{Suc} \end{cases}$$
$$d[x +_{\mathbb{N}} 1] := \begin{cases} \tilde{\varphi}(\bar{a}^{(m)}, b, \bar{0}^{(k)}, c[x +_{\mathbb{N}} 1]) & \text{if } c \in \mathrm{Lim} \\ \tilde{\varphi}(\bar{a}^{(m)}, b[x +_{\mathbb{N}} 1], \bar{0}^{(k+1)}) & \text{if } c = 0 \text{ and } b \in \mathrm{Lim} \\ \tilde{\varphi}(\bar{a}^{(m)}, b[x], d[x], \bar{0}^{(k)}) & \text{otherwise, i.e., } c \in \mathrm{Suc} \\ & \quad \text{or } (c = 0 \text{ and } b \in \mathrm{Suc}) \end{cases}$$

  Note that $m \neq 0$ implies that $a_1 \neq 0$ holds.

Remark A.18. Given $d = \phi \bar{a}^{(m+1)} \bar{0}^{(k)} \in \mathrm{OT}$ with $a_{m+1} \in \mathrm{Lim}$, we cannot expect that $\phi \bar{a}^{(m)} b \bar{0}^{(k)} \in \mathrm{OT}$ holds for every $b \prec a_{m+1}$, i.e., we cannot expect $\phi \bar{a}^{(m)} (a_{m+1}[x]) \bar{0}^{(k)} \in \mathrm{OT}$ to hold for any x. Take for instance $d := \phi a$ with $a := \phi \tilde{\varepsilon}_0 \tilde{1}$. Since $a \in \mathrm{Lim}$ holds, we have $d[x] = \tilde{\varphi}(a[x])$ with $a[x] = \tilde{\omega}^{\tilde{\varepsilon}_0} \tilde{\cdot} (x +_{\mathbb{N}} 1) = \tilde{\varepsilon}_0 \tilde{\cdot} (x +_{\mathbb{N}} 1)$. Hence, we have $\phi(a[0]) = \phi(\tilde{\varepsilon}_0) = \phi(\phi \tilde{1} 0) \notin \mathrm{OT}$ because of $\phi(\phi \tilde{1} 0) \prec_{\mathbf{lex}} \phi \tilde{1} 0$ and the definition of OT, and hence $(\tilde{\varphi}(a))[x] = \tilde{\varphi}(a[x]) \neq \phi(a[x])$ holds.

Theorem A.19.

(a) $\mathsf{TID} \vdash \forall d, x (d \in \mathrm{Suc} \rightarrow d[x] \prec d)$.

(b) $\mathsf{TID} \vdash \forall d, x (d \in \mathrm{Lim} \rightarrow (0 \prec d[x] \wedge d[x] \prec d[x +_{\mathbb{N}} 1] \wedge d[x] \prec d))$.

(c) $\mathsf{TID} \vdash \forall d, d_0 (d \in \mathrm{Lim} \wedge d_0 \prec d \rightarrow \exists x (d_0 \prec d[x]))$.

Proof. See [Ran15]. □

Corollary A.20. *Let $k, m \in \mathbb{N}$. TID proves that for every $\bar{a}^{(m)}, b, d_0 \in \mathrm{OT}$ with*
$$d_0 \prec \tilde{\varphi}(\bar{a}^{(m+1)}, \bar{0}^{(k)}, b)$$
the following holds:

(a) $b \in \mathrm{Lim} \rightarrow \exists x \big(d_0 \prec \tilde{\varphi}(\bar{a}^{(m+1)}, \bar{0}^{(k)}, b[x])\big)$.

(b) $(b \notin \mathrm{Lim} \wedge a_1 = 0 \wedge \ldots \wedge a_{m+1} = 0) \rightarrow \exists x \big(d_0 \prec \tilde{\omega}^{b[x]} \mathbin{\tilde{\cdot}} (x +_{\mathbb{N}} 1)\big)$.

(c) $(b = 0 \wedge a_{m+1} \in \mathrm{Lim}) \rightarrow \exists x \big(d_0 \prec \tilde{\varphi}(\bar{a}^{(m)}, a_{m+1}[x], \bar{0}^{(k+1)})\big)$.

Proof. See [Ran15]. □

References

[AR10] Bahareh Afshari and Michael Rathjen. A note on the theory of positive induction, ID_1^*. *Archive for Mathematical Logic*, 49:275–281, 2010. 10.1007/s00153-009-0168-9.

[Bee85] Michael J. Beeson. *Foundations of Constructive Mathematics: Metamathematical studies*. Springer Verlag, Berlin, Heidelberg, New York, 1985.

[BFPS81] Wilfried Buchholz, Solomon Feferman, Wolfram Pohlers, and Wilfried Sieg. *Iterated Inductive Definitions and Subsystems of Analysis: Recent Proof-Theoretical Studies*, volume 897 of *Lecture Notes in Mathematics*. Springer Verlag, Berlin, Heidelberg, New York, 1981.

[Bri75] Jane Bridge. A simplification of the Bachmann method for generating large countable ordinals. *The Journal of Symbolic Logic*, 40(2):171–185, 1975.

[BS88] Wilfried Buchholz and Kurt Schütte. *Proof Theory of Impredicative Subsystems of Analysis*, volume 2 of *Studies in Proof Theory Monographs*. Bibliopolis, Napoli, 1988.

[Buc05] Wilfried Buchholz. Prädikative Beweistheorie (Predicative proof theory). Lecture notes, University of Munich, 2004–2005.

[Buc16] Wilfried Buchholz. A survey on ordinal notations around the Bachmann-Howard ordinal. In Reinhard Kahle, Thomas Strahm, and Thomas Studer, editors, *Advances in Proof Theory*. Birkhaeuser, Springer Basel, 2016. To appear.

[Can86] Andrea Cantini. On the relation between choice and comprehension principles in second order arithmetic. *The Journal of Symbolic Logic*, 51(2):360–373, 1986.

[Fef92] Solomon Feferman. Logics for termination and correctness of functional programs, II. Logics of strength PRA. In Peter Aczel, Harold Simmons, and Stanley S. Wainer, editors, *Proof Theory*, pages 195–225. Cambridge University Press, 1992.

[FJS] Solomon Feferman, Gerhard Jäger, and Thomas Strahm. *Foundations of Explicit Mathematics*. Book in preparation.

[Jäg05] Gerhard Jäger. Metapredicative and explicit Mahlo: a proof-theoretic perspective. In Rene Cori, Alexander Razborov, Stevo Todorcevic, and Carol Wood, editors, *Proceedings of Logic Colloquium '00*, volume 19 of *Association of Symbolic Logic Lecture Notes in Logic*, pages 272–293. AK Peters, 2005.

[JKSS99] Gerhard Jäger, Reinhard Kahle, Anton Setzer, and Thomas Strahm. The proof-theoretic analysis of transfinitely iterated fixed point theories. *The Journal of Symbolic Logic*, 64(1):53–67, 1999.

[JS99] Gerhard Jäger and Thomas Strahm. Bar induction and ω model reflection. *Annals of Pure and Applied Logic*, 97(1–3):221–230, 1999.

[JS05] Gerhard Jäger and Thomas Strahm. Reflections on reflections in explicit mathematics. *Annals of Pure and Applied Logic*, 136(1–2):116–133, 2005. Festschrift on the occasion of Wolfram Pohlers' 60th birthday.

[Pro06] Dieter Probst. The proof-theoretic analysis of transfinitely iterated quasi least fixed points. *The Journal of Symbolic Logic*, 71(3):721–746, 2006.

[Pro16] Dieter Probst. *Modular Ordinal Analysis of Subsystems of Second-Order Arithmetic of Strength up to the Bachmann-Howard Ordinal.* Habilitation, Universität Bern, 2016. In preparation.

[Ran15] Florian Ranzi. *From a Flexible Type System to Metapredicative Wellordering Proofs.* PhD thesis, Universität Bern, 2015. `http://www.iam.unibe.ch/ltgpub/2015/ran15.pdf`.

[Rat92] Michael Rathjen. Fragments of Kripke–Platek set theory with infinity. In P. Aczel, H. Simmons, and S.S. Wainer, editors, *Proof Theory. A selection of papers from the Leeds Proof Theory Programme 1990*, pages 251–273. Cambridge University Press, 1992.

[RW93] Michael Rathjen and Andreas Weiermann. Proof-theoretic investigations on Kruskal's theorem. *Annals of Pure and Applied Logic*, 60:49–88, 1993.

[Sch54] Kurt Schütte. Kennzeichnung von Ordnungszahlen durch rekursiv erklärte Funktionen. *Mathematische Annalen*, 127:15–32, 1954. 10.1007/BF01361109.

[Sch92] Kurt Schütte. Beziehungen des Ordinalzahlensystems $\mathrm{OT}(\vartheta)$ zur Veblen-Hierarchie. Unpublished notes, 1992.

[Sim09] Stephen G. Simpson. *Subsystems of Second Order Arithmetic.* Cambridge University Press, second edition, 2009. Cambridge Books Online.

[Str99] Thomas Strahm. First steps into metapredicativity in explicit mathematics. In S. Barry Cooper and John K. Truss, editors, *Sets and Proofs*, volume 258 of *London Mathematical Society Lecture Notes*, pages 383–402. Cambridge University Press, 1999.

[vdMRW14] Jeroen van der Meeren, Michael Rathjen, and Andreas Weiermann. An order-theoretic characterization of the Howard-Bachmann-hierarchy, 2014. Preprint, `http://arxiv.org/abs/1411.4481`.