Semantic characterizations of number theories

Daniel Leivant

August 29, 1990

CMU-CS-90-169

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

To appear in Yiannis Moschovakis (ed.),
Logic from Computer Science,
Springer-Verlag, Berlin, 1990.

Abstract

We show that a number-theoretic formula is a theorem of First-Order Arithmetic if and only if it is true, as a statement about numbers, in all Henkin-structures that are closed under abstract jump (i.e. strict-Π^1_1 definitions), and that a number-theoretic formula is a theorem of Arithmetic with existential induction if and only if it is true in all Henkin-structures that contain their abstract RE (i.e. strict-Π^1_1 definable) sets.

Research partially supported by DARPA (DOD) and monitored by the Avionics Laboratory, Air Force Wright Aeronautical Laboratories, Aeronautical Systems Division (AFSC), Wright-Patterson AFB, Ohio 45433-6543, under Contract F33615-87-C-1499, ARPA Order Number 4976, Amendment 20; and by ONR grant N00014-84-K-0415. The view and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of ONR, DARPA, or the U.S. Government.

Keywords: First-order arithmetic, primitive recursive arithmetic, second-order logic, comprenesion principles, Henkin-structures, semantic characterization, interpretation, computational formulas, strict-Π^1_1 formulas, jump, König's Lemma.

Semantic characterizations of number theories

Daniel Leivant[1]

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
leivant@cs.cmu.edu

Dedicated to Leon Henkin
on the occasion of the 40'th anniversary of his
Completeness in the theory of types

The first-order predicate calculus is complete for its intended semantics, by Gödel's Completeness Theorem. Type Theory, though not complete for its intended semantics, is complete for the more liberal yet natural semantics of Henkin-structures. We derive here similar semantic characterizations for number theories.[2] We show that Peano's Arithmetic is sound and complete for truth (with object variables interpreted as natural numbers) in Henkin-structures that are closed under abstract jump (Theorem 33). By "abstract jump" we mean here Barwise's strict-Π^1_1 definability, a notion that agrees with Turing jump over the natural numbers, and which we argue is of foundational significance. We also show that Σ_1-Arithmetic is sound and complete for validity in all Henkin-structures that contain their "abstract RE" (i.e. strict-Π^1_1 definable) sets (Theorem 34). The paper is a refinement (of both results and exposition) of [Lei90].

1. Preliminaries

1.1. Number Theories

The vocabulary of First-Order Arithmetic, V_{FA}, has identifiers for zero and for all primitive recursive functions. The standard V_{FA}-structure has the set of natural numbers as universe, and the intended interpretations for 0 and for function identifiers.

[1] Research partially supported by ONR grant N00014-84-K-0415 and by DARPA grant F33615-87-C-1499.

[2] Of course, every first-order theory **T** has trivially a circular semantic characterization: **T** consists of the formulas valid in the models of **T**.

First-Order Arithmetic, **FA**, is a V_{FA}-theory, with axioms for equality, primitive recursive definitions, and induction. The *equality axioms* are $\neg(0=1)$, $\forall x.\, x = x$, $\forall x, y, z.\, x = y \rightarrow x = z \rightarrow y = z$, and, for each function identifier f and each $i \leq r = arity(f)$,

$$\forall x_1, \ldots, x_r, y.\, x_i = y \rightarrow f(x_1, \ldots, x_r) = f(x_1, \ldots, x_{i-1}, y, x_{i+1}, \ldots, x_r).$$

The *induction axioms* of **FA** are the instances of induction for all (first-order) V_{FA}-formulas. The *primitive recursive definitions* of **FA** are the universal closures of defining equations for all primitive recursive functions.[3] We write **PR** for the set of these formulas. For a primitive recursive function f, we let *degree*(f) denote the length of the (shortest in **PR**) primitive recursive definition of f.

It is easy to see that **FA** is a conservative extension of Peano's Arithmetic. Note that Peano's third axiom follows from $\neg(0=1)$ by induction, and the fourth axiom follows from the defining equations for the predecessor function.

Σ_1-Arithmetic, $\mathbf{\Sigma_1 A}$, is like **FA** except that induction is restricted to existential V_{FA}-formulas. Computationally, $\mathbf{\Sigma_1 A}$ is the same as Primitive Recursive Arithmetic, **PRA** (which has induction only for V_{FA}-equations), since they prove the same Π^0_2 formulas, that is, they have the same provably recursive functions [Par77].

It will be useful to identify conditions for doing away with the axiom $\neg(0 = 1)$. For a number theory **A**, let $\mathbf{A}^-$ denote **A** without $\neg(0=1)$.

LEMMA 1 *Let φ be a V_{FA}-formula in which equality has no negative occurrences, and let* **A** *be one of the number theories above. If* $\mathbf{A} \vdash \varphi$, *then* $\mathbf{A}^- \vdash \varphi$.

PROOF: Troelstra observed [Tro73] that if a formula φ is provable in **A**, then the result of replacing (hereditarily) in φ every negated subformula $\neg\psi$ by $\psi \rightarrow 0=1$ is a theorem of $\mathbf{A}^-$. Let φ' be a prenex-disjunctive normal form for φ, so the equivalence $\varphi \leftrightarrow \varphi'$ is provable in first-order logic. Then $\mathbf{A} \vdash \varphi$ implies $\mathbf{A} \vdash \varphi'$, from which $\mathbf{A}^- \vdash \varphi'$, since φ' is without negation, whence $\mathbf{A}^- \vdash \varphi$. □

Let *neq* be the primitive recursive characteristic function of inequality: $neq(x, y) = 0$ iff $x \neq y$. For a V_{FA}-formula φ, let $\tilde{\varphi}$ be the result of replacing in φ each negatively occurring equation, $t = s$, by $\neg(neq(t, s) = 0)$. Then $\tilde{\varphi}$ has no negative occurrences of equality, and we have, immediately from the definitions, $\mathbf{PRA} \vdash \varphi \leftrightarrow \tilde{\varphi}$.

Combined with Lemma 1 this implies

[3] Our development remains valid if we expand the set of functions to include all partial recursive functions, where the defining equations are Herbrand-Gödel functional programs, as in [Kle52], provably coherent in Primitive Recursive Arithmetic. A Herbrand-Gödel program $\mathcal{P}$ is *coherent* if its operational semantics generates a single-valued relation. Coherence is an undecidable property, but there is a collection C of functional programs such that membership in C is decidable in real time, and such that every partial recursive function has a program in C. The programs obtained from any one of the standard proofs of this fact (as e.g. in [Kle52]) are all provably coherent in Primitive Recursive Arithmetic.

LEMMA 2 *Let φ be a V_{FA}-formula, and let* $\mathbf{A}$ *be one of the number theories above. If* $\mathbf{A} \vdash \varphi$ *then* $\mathbf{A} \vdash \varphi \leftrightarrow \tilde{\varphi}$ *and* $\mathbf{A}^- \vdash \tilde{\varphi}$.

1.2. Henkin-structures

The *language of second-order logic* is an extension of the language of first-order logic (with equality and function identifiers) with relational variables of all finite arities, and with quantification over them. Our basic proof-system for second-order logic, $\mathbf{SOL}_0$, is obtained from the first-order predicate calculus with equality by treating relational variables in par with object variables (without comprehension); see for example [Pra65, §V.1] for details. Given a class Φ of formulas, Φ-*Comprehension* is the schema $\exists R \forall \vec{x} (R(\vec{x}) \leftrightarrow \varphi)$, where R is a relational variable that does not occur free in φ, $arity(R) = arity(\vec{x})$, and $\varphi \in \Phi$. If Φ is a collection of second-order formulas, we write $\mathbf{SOL}(\Phi)$ for $\mathbf{SOL}_0$ augmented with Φ-comprehension. $\mathbf{SOL}$ will denote $\mathbf{SOL}_0$ with comprehension for all (second-order) formulas.

Since the collection of second-order formulas that are valid (under the standard interpretation of relational quantification) is not in the arithmetical hierarchy, let alone effectively enumerable, even $\mathbf{SOL}$ is incomplete for standard validity. However, second-order logic is complete for the broader class of Henkin-structures [Hen50]. A *Henkin-structure*, $\mathcal{H}$, consists of a first-order structure over some universe A, augmented with, for each $r \geq 1$, a collection $\mathcal{H}_r$ of r-ary relations over A. Semantic satisfaction, $\mathcal{H} \models \varphi$, is defined using $\mathcal{H}_r$ as the range of quantifiers over r-ary relations. If Φ is a class of second-order formulas, then $\mathcal{H}$ is *closed under* Φ if, for each $\varphi \equiv \varphi[\vec{x}, \vec{R}]$ in Φ, with free object variables $\vec{x} = (x_1 \ldots x_k)$, and free relational variables $\vec{R} = (R_1 \ldots R_l)$ (with $r_i = \text{arity}(R_i)$), and all $Q_1 \in \mathcal{H}_{r_1}, \ldots, Q_l \in \mathcal{H}_{r_l}$, the set $\{(a_1 \ldots a_k) \in A^k \mid \mathcal{H} \models \varphi[\vec{a}/\vec{x}, \vec{Q}/\vec{R}]\}$ is in $\mathcal{H}_k$.

The proof in [Hen50] establishes the following.

THEOREM 3 [Henkin] *Let Φ be a class of second-order formulas. A second-order formula is valid in all Henkin-structures closed under Φ iff it is provable in* $\mathbf{SOL}(\Phi)$.

1.3. Computational formulas

A second-order formula is *computational* if it is of the form $\forall \vec{R} \exists \vec{x} \psi$, where $\vec{R}$ are relational variables and ψ is quantifier-free, i.e. if it is *strict*-Π^1_1 in the sense of [Bar69, Bar75]. A computational formula without free relational variables is *relationally closed* (*r-closed* for short). $\mathbf{Comp}$ (respectively, $\mathbf{Comp}_0$) will denote the set of formulas equivalent in $\mathbf{SOL}_0$ to a computational formula (respectively, to an r-closed computational formula).

The term "computational" is induced by the fact that each computational formula describes a computation process, which becomes apparent when the formula is converted into an equivalent "computational normal formula," as follows. Let V be a vocabulary, and fix a tuple $\vec{R}$ of relational identifiers. Let χ be a syntactic parameter for quantifier-free V-formulas. A *computational normal formula* is a formula of the form

$$\varphi \equiv \forall R_1 \dots R_r \, [\, \forall \vec{u} \, (\, \iota_1 \wedge \iota_2 \cdots \wedge \kappa_1 \wedge \cdots) \;\rightarrow\; \exists \vec{v} \, (\, \theta_1 \vee \cdots) \,],$$

where each ι_n is the disjunction of formulas of the form $\chi \rightarrow R_i(\vec{x})$, each κ_n is the disjunction of formulas of the form $\chi \wedge R_j(\vec{y}) \rightarrow R_i(\vec{x})$, and each θ_n is the conjunction of formulas of one the forms χ or $R(\vec{x})$. The formula φ states (about its free variables) that every process that initializes the values of $\vec{R}$ as prescribed by the ι_n's, and inductively closes these relations as prescribed by the κ_n's, will reach values that satisfy some "target condition" θ_n.

Each computational normal formula defines, uniformly for all V-structures, the operational semantics of a certain finite state machine (see [Lei89] for details). The connection with computational formulas is given by the straightforward observation that every computational formula is equivalent, in $\mathbf{SOL}_0$, to a computational normal formula.

The significance of computational formulas is further manifest in the following.

THEOREM 4 [Kreisel] *Every computational V_{FA}-formula is equivalent in the standard V_{FA}-structure to an existential formula.*

Hence, every r-closed computational V_{FA}-formula defines in the standard V_{FA}-structure an RE set.

PROOF: Using familiar sequence-coding, every computational formula is equivalent in the standard V_{FA}-structure to a computational formula of the form $\forall R \, \exists x \, \psi$, where R is unary, and $\psi \equiv \psi[R]$ is quantifier-free with no free variables other than x and u. Let

$$m_\psi(x,u) =_{\text{df}} \max\{ \text{ value of } t[x,u] \mid R(t) \text{ is a subformula of } \psi\}.$$

Recall that König's Lemma states that every infinite finitely-branching tree has an infinite branch. This implies that for any formula $\varphi \equiv \varphi[R]$ with a single set variable R,

$$\forall R \, \exists m \, \varphi[R_{\leq m}] \;\rightarrow\; \exists h \, \forall R \, \exists m \leq h. \; \varphi[R_{\leq m}].$$

(Here $R_{\leq m} =_{\text{df}} R \cap \{0, \dots, m\}$.)

We have

$$\begin{aligned}
\forall R \, \exists x \, \psi[R] \;&\leftrightarrow\; \forall R \, \exists x \, \exists m \, \psi[R_{\leq m_\psi(x,u)}] \\
&\leftrightarrow\; \forall R \, \exists m \, \exists x \, (\, m = m_\psi(x,u) \wedge \psi[R_{\leq m}] \,) \\
&\leftrightarrow\; \exists h \, \forall \sigma \subseteq \{0, \dots, h\} \; \exists m \leq h \, \exists x \, (\, m = m_\psi(x,u) \wedge \psi[\sigma_{\leq m}] \,).
\end{aligned}$$

The forward direction of the last equivalence holds by König's Lemma, and the backward direction is straightforward. The latter formula is equivalent in the standard structure to an existential formula, since the universal quantifier is bounded. $\square$

The proof above of the equivalence of computational formulas to existential formulas clearly applies to any countable admissible structure [Bar69, Bar75 (Theorem VIII.3.1)]. However, this equivalence fails to hold in general for structures which do not contain a code for every completed computation over elements of the structure. For example, if $V_s = \{0, \mathrm{s}\}$, then it is easy to see that every RE set of natural numbers is defined in the standard V_s-structure[4] by a computational formula (compare Lemma 14 below), whereas the sets defined in the standard V_s-structure by existential formulas, or even by first-order formulas, are all recursive. (In fact, even for the vocabulary $V_+ = \{0, \mathrm{s}, +\}$, every set of natural numbers defined in the standard V_+-structure by a first-order formula is recursive, by [Pre30].) Thus, computational formulas might be regarded as the appropriate generalization to all structures of recursive enumerability; they reduce to existential formulas over structures in which computations are representable internally, but are in general stronger.

Of interest is also the strength of computational formulas as queries. A k-ary *query* (or *global relation*) over a class $\mathcal{C}$ of structures assigns to each structure $\mathcal{S} \in \mathcal{C}$ a k-ary relation over the universe $|\mathcal{S}|$ of $\mathcal{S}$. If $\mathcal{C}$ consists of V-structures, and φ is a V-formula whose free variables are among $u_1 \dots u_k$, then $\lambda u_1 \dots u_k.\ \varphi$ determines a query over $\mathcal{C}$, that assigns to $\mathcal{S} \in \mathcal{C}$ the relation $\{(a_1 \dots a_k) \in |\mathcal{S}|^k \mid \mathcal{S} \models \varphi[\vec{a}/\vec{u}]\}$. Now, over ordered finite structures the queries defined by computational formulas are exactly the co-NP queries [Fag74, JS74], whereas the queries defined even by all first-order formulas, are a strict subclass of the queries computable in deterministic log-space[5]. The significance of computational formulas as query definitions has found applications in Descriptive Computational Complexity [Lei89], in Finite Model Theory [KV87, KV88], and in Logics of Programs [Lei85, Lei85a].

Note that if a computational formula φ has free relational variables $\vec{Q}$, then it determines a computational process that uses $\vec{Q}$ as oracles, and is equivalent over countable admissible structures to an existential formula with $\vec{Q}$ free. Thus, φ defines an abstract notion of relative RE, that is — an abstract form of Kleene's jump.

2. Direct second-order interpretation of number theories

In this section we define a second-order interpretation of V_{FA} which is direct, in the sense that the target formalism has defining equations for primitive recursive functions, in contrast to the "full" interpretation we define in the sequel. While the full interpretation is more logically prestine, the direct interpretation is easier to formulate and verify.

[4] where 0 is interpreted as zero and s as the successor function

[5] [Fag75] proves that graph connectivity is not a first-order definable query; an elegant simple proof of this can be found in [GV85]. [Imm87] observes that all first-order queries over finite ordered structures are computable in deterministic log-space.

2.1. Definition of the direct interpretation

Let

$$N[x] \;:\equiv\; \forall R(\, Cl[R] \;\rightarrow R(x)\,),$$

where

$$Cl[R] \;:\equiv\; \forall u\,(R(u) \rightarrow R(\mathrm{s}(u))\;\wedge\; R(0).$$

Note that $N \in \mathbf{Comp}_0$. If $\mathcal{S}$ is a structure that satisfies $\forall u\, \neg(\mathrm{s}(u)=0)$ and $\forall u, v\,(\mathrm{s}(u) = \mathrm{s}(v) \;\rightarrow\; u = v\,)$, then the extension of N in $\mathcal{S}$ is a copy of the natural numbers.

If $\vec{t} = (t_1 \dots t_r)$ is a tuple of terms, we write $N[\vec{t}]$ for $N[t_1] \wedge \cdots \wedge N[t_r]$. If $\alpha_1, \dots, \alpha_k$ are formulas or terms, we write $var(\alpha_1, \dots, \alpha_k)$ for the set of variables that occur free in $\alpha_1, \dots, \alpha_k$.

If φ is a V_{FA}-formula, then φ^N denotes φ with quantifiers relativized to N. Assuming $N[x]$ we get induction with respect to x for a formula φ, by instantiation of the universal relational quantifier to the relation $\lambda x.\varphi$:

$$\forall u\,(\varphi[u/x] \rightarrow \varphi[\mathrm{s}(u)/x]) \;\wedge\; \varphi[0/x] \;\rightarrow \varphi.$$

This is legitimate by comprehension for φ. However, if φ is a first-order V_{FA}-formula, then φ^N is in general not first-order (because N is not), so first-order comprehension does not yield induction for the interpretation of first-order V_{FA}-formulas!

We define the *direct interpretation* of V_{FA} as having N as the formula defining the target universe, and with the V_{FA}-identifiers interpreted by themselves.

Given a formalism $\mathbf{C}$ for second-order (or higher-order) logic (with constant and function identifiers), the *directly-interpreted number theory of* $\mathbf{C}$ is

$$\begin{aligned} \mathbf{DNT}[\mathbf{C}] \;&=_{\mathrm{df}}\; \{\varphi \mid \varphi \text{ is a closed } V_{FA}\text{-formula, and } \;\; \mathbf{C},\, \mathbf{PR},\, \neg(0=1) \vdash \varphi^N\,\} \\ &=\; \{\varphi \mid \varphi \text{ is a closed } V_{FA}\text{-formula, and } \;\; \mathbf{C},\, \mathbf{PR} \vdash \tilde{\varphi}^N\,\}. \end{aligned}$$

It is not hard to delineate the direct number theory of (impredicative) Second-Order Logic, **SOL**. Recall that Impredicative Analysis, i.e. Second-Order Arithmetic, is the extension of **FA** with quantification over relations, with induction formulated as a single axiom, $\forall x.N[x]$, and with comprehension for all formulas in the language. The following is essentially due to Prawitz [Pra65].

THEOREM 5 *A first-order V_{FA}-formula φ is a theorem of Impredicative Analysis iff φ^N is provable in* $\mathbf{SOL} + \mathbf{PR} + \neg(0=1)$, *i.e. iff $\tilde{\varphi}^N$ is provable in* $\mathbf{SOL} + \mathbf{PR}$.

2.2. Correctness of the direct interpretation

The following proposition states the correctness of the direct interpretation of V_{FA} in $\mathbf{SOL}(\mathbf{Comp}_0)$.

PROPOSITION 6 *For every V_{FA}-identifier f,*

$$\mathbf{SOL}(\mathbf{Comp}_0) \vdash N[\vec{x}] \rightarrow N[f(\vec{x})]$$

(where $arity(\vec{x}) = arity(f)$).

PROOF: By induction on *degree*(f). If f is one of the initial functions, then the proposition is trivial. The case where f is defined by composition is straightforward.

Suppose f is defined by recurrence,

$$\begin{aligned} f(0,\vec{u}) &= g(\vec{u}) \\ f(\mathrm{s}(v),\vec{u}) &= h(v,\vec{u},f(v,\vec{u})). \end{aligned}$$

Arguing within $\mathbf{SOL}(\mathbf{Comp}_0)$ we have, by induction assumption,

$$N[\vec{u}] \rightarrow N[g(\vec{u})] \tag{1}$$

and

$$\forall w.\, N[v,\vec{u},w] \rightarrow N[h(v,\vec{u},w)]. \tag{2}$$

Assume

$$N[v,\vec{u}]. \tag{3}$$

From $N[v] \equiv \forall R.Cl[R] \rightarrow R(v)$, instantiating $R(x)$ to the r-closed computational formula $N[f(x,\vec{u})]$, we get

$$\forall x\,(N[f(x,\vec{u})] \rightarrow N[f(\mathrm{s}(x),\vec{u})]) \wedge N[f(0,\vec{u})] \rightarrow N[f(v,\vec{u})]. \tag{4}$$

By (2) and (3) $N[f(x,\vec{u})]$ implies $N[h(x,\vec{u},f(x,\vec{u}))]$, which by the second defining equation for f yields $N[f(\mathrm{s}(x),\vec{u})]$. This proves the first conjunct in (4). The second conjunct is immediate from (1) and the first defining equation for f. Thus we get the conclusion of (4), $N[f(v,\vec{u})]$, based on assumption (3), which is precisely the statement of the proposition. □

2.3. Soundness of the direct interpretation

In this section we prove that the direct interpretation of **FA** is sound for $\mathbf{SOL}(\mathbf{Comp})$, that is, if $\mathbf{FA} \vdash \varphi$ then $\mathbf{SOL}(\mathbf{Comp}), \mathbf{PR}, \neg(0=1) \vdash \varphi$. Analogously, we show that the interpretation of $\mathbf{\Sigma_1 A}$ is sound for $\mathbf{SOL}(\mathbf{Comp}_0)$.

LEMMA 7 **Comp** *and* $\mathbf{Comp}_0$ *are closed under conjunction and disjunction.*

PROOF: Trivial, by basic quantifier rules. □

LEMMA 8 $\mathbf{SOL_0} \vdash Cl[N]$.

PROOF: Straightforward. $\square$

LEMMA 9 *A formula φ of the form $\exists x^N \psi$, where ψ is [r-closed] quantifier-free, is equivalent in* $\mathbf{SOL(Comp_0)}$ *to an [r-closed] computational formula.*

PROOF: We have

$$\varphi \;\equiv\; \exists x\,(\,\psi \wedge \forall R\,(\,Cl[R] \rightarrow R(x)\,)\,)$$

(assuming, without loss of generality, that R does not occur in ψ). We claim that this is equivalent to

$$\varphi' \;\equiv_{\mathrm{df}}\; \forall R\,\exists x\,(\,\psi \wedge (\,Cl[R] \rightarrow R(x)\,)\,),$$

which is clearly an r-closed computational formula. φ implies φ' trivially. For the converse, instantiating R in φ' to N, yields $\exists x\,(\,\psi \wedge (\,Cl[N] \rightarrow N[x]\,)\,)$, which by Lemma 8 implies φ.
$\square$

LEMMA 10 *For every V_{FA}-formula φ, comprehension for φ^N is provable in* $\mathbf{SOL(Comp)}$.

PROOF: By induction on φ. Without loss of generality, we assume that $\wedge$, $\neg$, and $\exists$ are the only logical constants. If φ is quantifier-free then it is computational, and the lemma is trivial.

If $\varphi \equiv \psi \wedge \chi$, then $\varphi^N \equiv \psi^N \wedge \chi^N$. By induction assumption $\mathbf{SOL(Comp)}$ proves $\exists P \forall \vec{u}\; (P(\vec{u}) \leftrightarrow \psi^N)$ and $\exists Q \forall \vec{v}\; (Q(\vec{v}) \leftrightarrow \chi^N)$, where $\vec{u}$ and $\vec{v}$ list $var(\psi)$ and $var(\chi)$, respectively. By comprehension for quantifier-free formulas

$$\forall P, Q\, \exists R \forall \vec{w}\,(\,R(\vec{w}) \leftrightarrow P(\vec{u}) \wedge Q(\vec{v})\,),$$

where $\vec{w}$ lists $var(\varphi) = \vec{u} \cup \vec{v}$. So

$$\mathbf{SOL(Comp)} \vdash \exists R \forall \vec{w}\,(\,R(\vec{w}) \leftrightarrow \varphi^N\,).$$

The case $\varphi \equiv \neg\psi$ is treated similarly.

Finally, if $\varphi \equiv \exists x\,\psi$, then $\varphi^N \equiv \exists x(N[x] \wedge \psi^N)$. By induction assumption, $\mathbf{SOL(Comp)}$ proves $\exists R \forall \vec{u}, x\,(\,R(\vec{u}, x) \leftrightarrow \psi^N\,)$, where $\vec{u}$ lists $var(\varphi)$. Hence, $\mathbf{SOL(Comp)}$ proves (using the same R), $\exists R \forall \vec{u}\,(\,(\exists x\,(N[x] \wedge R(\vec{u}, x))) \leftrightarrow \varphi^N)$. By Lemma 9 $\mathbf{SOL(Comp)}$ proves $\exists Q \forall \vec{u}\;(\,Q(\vec{u}) \leftrightarrow \exists x\,(\,N[x] \wedge R(\vec{u}, x))\,)$, i.e., $\exists Q \forall \vec{u}\,(\,Q(\vec{u}) \leftrightarrow \varphi^N\,)$. $\square$

LEMMA 11 *For each V_{FA}-term t,*

$$\mathbf{SOL(Comp_0)} \vdash N[var(t)] \rightarrow N[t].$$

PROOF: By induction on (the structure of) t. The basis is trivial, and the induction step is Proposition 6. $\square$

LEMMA **12** *Let φ be a V_{FA}-formula.*

1. *If* $\mathbf{FA}^- \vdash \varphi$, *then* $\mathbf{SOL(Comp)}, \mathbf{PR} \vdash N[var(\varphi)] \rightarrow \varphi^N$.
2. *If* $\mathbf{\Sigma_1 A}^- \vdash \varphi$, *then* $\mathbf{SOL(Comp_0)}, \mathbf{PR} \vdash N[var(\varphi)] \rightarrow \varphi^N$.

PROOF: By induction on proofs of $\mathbf{\Sigma_1 A}$ and $\mathbf{FA}$, say in the Hilbert-style deductive calculus of [Kle52, §19] for the logical constants $\neg, \rightarrow$, and $\forall$.

The axioms are of the following types.

1. φ is an instance of a propositional schema. Then φ^N is an instance of the same propositional schema.
2. φ is of the form $\forall x \psi \rightarrow \psi[t/x]$. Then $N[var(\forall x \psi), t] \rightarrow \varphi^N$ outright. By Lemma 11, $N[var(t)] \rightarrow N[t]$. Since $var(t) \subseteq var(\varphi)$, we get $N[var(\varphi)] \rightarrow \varphi^N$.
3. φ is one of the logical axioms for equality. Then φ is in $\mathbf{SOL}_0$.
4. $\varphi \in \mathbf{PR}$, trivial.
5. φ is an instance of Induction, $\forall x.\, Cl[\psi] \rightarrow \psi[x]$, with ψ existential for (2). Then $\varphi^N \equiv \forall x^N (Cl[\psi]^N \rightarrow \psi^N[x])$. We have comprehension for the formula $N[u] \wedge \psi^N[u]$: by Lemma 10 for (1), and by Lemma 9 for (2). Therefore, $N[x]$ implies

$$\forall u\,((N[u] \wedge \psi^N[u]) \rightarrow (N[\mathrm{s}(u)] \wedge \psi^N[\mathrm{s}(u)])) \;\wedge\; (N[0] \wedge \psi^N[0]) \;\rightarrow\; (N[x] \wedge \psi^N[x]),$$

which clearly implies

$$\forall u^N (\psi^N[u] \rightarrow \psi^N[\mathrm{s}(u)]) \wedge \psi^N[0] \;\rightarrow\; \psi^N[x],$$

i.e., $Cl[\psi]^N \rightarrow \psi^N[x]$. We have thus proved φ^N.

For the induction step of the proof, we check that the statement of the lemma is preserved under the two inference rules:

Detachment: φ is derived from $\chi \rightarrow \varphi$ and χ. By induction assumption, $N[var(\chi, \varphi)] \rightarrow (\chi^N \rightarrow \varphi^N)$ and $N[var(\chi)] \rightarrow \chi^N$ are both provable (in $\mathbf{SOL(Comp)}$ for part (1) of the lemma and in $\mathbf{SOL(Comp_0)}$ for (2)). Let χ_0 be χ with 0 substituted for all free occurrences of variables not free in φ. Then $N[var(\varphi)] \rightarrow (\chi_0^N \rightarrow \varphi^N)$ and $N[var(\varphi)] \rightarrow \chi_0^N$ are both provable. Thus $N[var(\varphi)] \rightarrow \varphi^N$ is provable, by Detachment.

Generalization: $\varphi \equiv \chi \to \forall x \psi$ is derived from $\chi \to \psi$. By induction assumption $N[var(\varphi), x] \to (\chi^N \to \psi^N)$ is provable. Therefore, $N[var(\varphi)] \to (\chi^N \to \forall x^N \psi^N)$ is provable, by Generalization, since x must not be free in χ.

This concludes the induction step and the proof. $\square$

From Lemmas 2 and 12 we obtain

THEOREM **13** *Let φ be a V_{FA}-formula. If* **FA** $\vdash \varphi$, *then*

$$\mathbf{SOL(Comp)}, \mathbf{PR}, \neg(0=1) \vdash N[var(\varphi)] \to \varphi^N, \text{ and}$$
$$\mathbf{SOL(Comp)}, \mathbf{PR} \vdash N[var(\varphi)] \to \tilde{\varphi}^N.$$

Similarly, if $\Sigma_1\mathbf{A} \vdash \varphi$, *then*

$$\mathbf{SOL(Comp_0)}, \mathbf{PR}, \neg(0=1) \vdash N[var(\varphi)] \to \varphi^N, \text{ and}$$
$$\mathbf{SOL(Comp_0)}, \mathbf{PR} \vdash N[var(\varphi)] \to \tilde{\varphi}^N.$$

3. Full second-order interpretations of number theories

We define an interpretation of the vocabulary of **SOL** that has each V_{FA}-identifier interpreted by an r-closed computational formula that defines its graph. The target formalism of the interpretation cannot make do with no constants at all, since to interpret 0 and s in the absence of constants we would need second-order constant-free formulas, φ with only x free, and ψ with only x and y free, such that $\mathbf{SOL} \vdash \exists! x\, \varphi$ and $\mathbf{SOL} \vdash \forall y\, (M[y] \to \exists! x\, \psi)$ where M is a formula interpreting N. Clearly, no such formulas exist. We therefore assume that 0 and s are present in the target vocabulary.

3.1. Graphs of primitive recursive functions

For each V_{FA}-identifier f, we define a formula G_f, by induction on *degree*(f), as follows.

- If f is the zero function, then $G_f[x, z] \equiv_{\mathrm{df}} (z = 0)$.
- If f is the successor function, then $G_f[x, z] \equiv_{\mathrm{df}} (z = \mathsf{s}(x))$.
- If f is the i'th-out-of-n projection function, then $G_f[x_1 \ldots x_n, z] \equiv_{\mathrm{df}} (z = x_i)$.
- If f is defined by composition, $f(\vec{x}) = h(g_1(\vec{x}), \ldots, g_k(\vec{x}))$, then

$$G[\vec{x}, z] \equiv_{\mathrm{df}} \exists y_1 \ldots y_k.\, G_{g_1}[\vec{x}, y_1] \wedge \cdots \wedge G_{g_k}[\vec{x}, y_k] \wedge G_h[\vec{y}, z].$$

- If f is defined by recurrence, $f(0,\vec{u}) = g(\vec{u})$, $f(\mathrm{s}(v),\vec{u}) = h(v,\vec{u},f(v,\vec{u}))$, then

$$\begin{aligned} G[\vec{x},z] \quad \equiv_{\mathrm{df}} \quad & \forall Q\,(\,\forall \vec{u},y\,(\,G_g[\vec{u},y] \rightarrow Q(0,\vec{u},y)\,) \\ & \qquad \wedge \forall \vec{u},v,w,y\,(\,Q(v,\vec{u},w) \wedge G_h[v,\vec{u},w,y] \rightarrow Q(\mathrm{s}(v),\vec{u},y)\,) \\ & \qquad \rightarrow Q(\vec{x},z)\,) \end{aligned}$$

(where $arity(\vec{x}) = arity(\vec{u}) + 1$).

We need the following generalization of Lemma 9.

LEMMA **14** *Let φ be a conjunction of formulas of the form $G_f[\vec{t}]$, formulas of the form $N[t]$, and quantifier-free [r-closed] formulas. Then $\exists\vec{x}\,\varphi$ is equivalent in* **SOL(Comp$_0$)** *to an [r-closed] computational formula.*

In particular, every formula of the form $G_f[\vec{t}]$ is equivalent in **SOL(Comp$_0$)** *to an r-closed computational formula.*

PROOF: Suppose

$$\varphi \;\equiv\; G_{f_1}[\vec{t}_1,s_1] \wedge \cdots \wedge G_{f_k}[\vec{t}_k,s_k] \wedge N[q_1] \wedge \cdots \wedge N[q_l] \wedge \alpha,$$

where α is quantifier-free. We prove the lemma by main induction on the number l of conjuncts $N[q_i]$, secondary induction on $m = \max_{i \leq k}[degree(f_i)]$, and ternary induction on the number n of conjuncts G_{f_i} with $degree(f_i) = m$.

If $l = m = 0$ (i.e. $k = 0$), then the lemma is trivial.

If $l > 0$, then

$$\begin{aligned} & \varphi \;\equiv\; \exists x\,(\,\psi \wedge \forall R.\,Cl[R] \rightarrow R(q_l)\,) \\ \text{where} \quad & \psi \;\equiv\; N[q_1] \wedge \cdots \wedge N[q_{l-1}] \wedge \alpha. \end{aligned}$$

As in the proof of Lemma 9, φ is equivalent to

$$\forall R\,\exists x\,(\,\psi \wedge (Cl[R] \rightarrow R(q_l))\,),$$

which is equivalent, by elementary quantifier rules, to

$$\varphi' \equiv_{\mathrm{df}} \forall R\,\exists x,u\,(\,\psi \wedge \beta\,),$$

where u is fresh and β is the quantifier-free formula $(\,R(u) \rightarrow R(\mathrm{s}(u))\,) \rightarrow R(q_l)$. By induction assumption, φ' is equivalent in **SOL(Comp$_0$)** to an r-closed computational formula.

If $k > 0$ (and $l = 0$), then φ is of the form $\exists x\,(\,\psi \wedge G_f[\vec{t},s]\,)$, where $degree(f) = m$. We proceed by cases for the definition of f. If f is initial, then $G_f[\vec{t},s]$ is quantifier-free, and we are done by induction assumption. If f is defined by composition, then $G_f[\vec{t}, s]$ is of the form

$$\exists v_1 \dots v_k \, (\, G_{g_1}[\vec{t}, v_1] \wedge \cdots \wedge G_{g_k}[\vec{t}, v_k] \wedge G_h[\vec{v}, s] \,),$$

with *degree* $(g_1), \dots,$ *degree* (g_k), *degree* $(h) < m$. So φ is equivalent to

$$\exists x, v_1 \dots v_k \; (\, \psi \wedge G_{g_1}[\vec{t}, v_1] \wedge \cdots \wedge G_{g_k}[\vec{t}, v_k] \wedge G_h[\vec{v}, s] \,),$$

for which the lemma holds by induction assumption.

Suppose that f is defined by recurrence.

CLAIM 1. $G_f[\vec{t}, s]$ *is r-closed computational.* By definition, $G_f[\vec{t}, s]$ is of the form $\forall Q. \chi$, where

$$\begin{aligned} \chi \;\; \equiv \;\; & \forall \vec{u}, v \, (\, G_g[\vec{u}, v] \to Q(0, \vec{u}, v) \,) \\ & \wedge \; \forall \vec{w}, y, z, a \, (\, Q(y, \vec{w}, z) \wedge G_h[y, \vec{w}, z, a] \to Q(\mathrm{s}(y), \vec{w}, a) \,) \\ & \to Q(\vec{t}, s). \end{aligned}$$

The formula χ is equivalent to

$$\begin{aligned} \chi' \;\; \equiv_{\mathrm{df}} \;\; & \exists \vec{u}, v \, (\, G_g[\vec{u}, v] \wedge \neg Q(0, \vec{u}, v) \,) \\ & \vee \; \exists \vec{w}, y, z, a \, (\, Q(y, \vec{w}, z) \wedge G_h[y, \vec{w}, z, a] \wedge \neg Q(\mathrm{s}(y), \vec{w}, a) \,) \\ & \vee \; Q(\vec{t}, s). \end{aligned}$$

Since *degree* (g), *degree* $(h) < m$, each one of the disjuncts of χ' is equivalent in $\mathbf{SOL}(\mathbf{Comp}_0)$ to an r-closed computational formula, by induction assumption, and therefore, by Lemma 7, $G_f[\vec{t}, s]$ is also equivalent to an r-closed computational formula, proving Claim 1.

CLAIM 2. φ *is equivalent, in* $\mathbf{SOL}(\mathbf{Comp}_0)$, *to*

$$\varphi' \;\; \equiv_{\mathrm{df}} \;\; \forall Q \, \exists x \, (\, \psi \wedge \chi' \,).$$

Clearly, $\varphi \equiv \exists x \, (\, \psi \wedge \forall Q \, \chi \,)$ implies $\forall Q \, \exists x \, (\, \psi \wedge \chi \,)$, since Q is not free in ψ. This implies $\forall Q \, \exists x \, (\, \psi \wedge \chi' \,)$, i.e. φ', since χ' is equivalent to χ.

For the converse, assume φ'. Then, by instantiating Q to $\lambda v, \vec{u}, w. \, G_f[v, \vec{u}, w]$, we get

$$\begin{aligned} \varphi_0 \;\; & \equiv_{\mathrm{df}} \;\; \exists x \, (\, \psi \wedge \chi'[G_f \, / \, Q] \,) \\ & \equiv \;\; \exists x \, (\, \psi \wedge \chi[G_f \, / \, Q] \,). \end{aligned}$$

This instantiation is legitimate in $\mathbf{SOL}(\mathbf{Comp}_0)$ by Claim 1. But the two premises of $\chi[G_f \, / \, Q]$, that is

$$\begin{aligned} & \forall \vec{u}, v \, (\, G_g[\vec{u}, v] \to G_f[0, \vec{u}, v] \,) \\ \text{and} \qquad & \forall \vec{w}, y, z, a \, (\, G_f[y, \vec{w}, z] \wedge G_h[y, \vec{w}, z, a] \to G_f[\mathrm{s}(y), \vec{w}, a] \,) \end{aligned}$$

are straightforward in $\mathbf{SOL}(\mathbf{Comp}_0)$. Therefore, $\chi[G_f \, / \, Q]$ implies its antecedent, i.e. $G_f[\vec{t}, s]$. So φ_0 implies, in $\mathbf{SOL}(\mathbf{Comp}_0)$, $\exists x \, (\, \psi \wedge G_f[\vec{t}, s] \,)$, i.e. φ. This proves Claim 2.

To prove the lemma it remains to show that $\varphi' \equiv \forall Q\, \exists x\, (\, \psi \wedge \chi'\,)$ is equivalent in **SOL(Comp$_0$)** to an r-closed computational formula. We have

$$\begin{aligned} \exists x\, (\, \psi \wedge \chi'\,) \;\equiv\;& \exists x, \vec{u}, v.\, (\, \psi \wedge G_g[\vec{u}, v] \wedge \neg Q(0, \vec{u}, v)\,) \\ & \vee\, \exists x, \vec{w}, y, z, a\, (\, \psi \wedge Q(y, \vec{w}, z) \wedge G_h[y, \vec{w}, z, a] \wedge \neg Q(\mathrm{s}(y), \vec{w}, a)\,) \\ & \vee\, \exists x\, (\, \psi \wedge Q(\vec{t}, s)\,). \end{aligned}$$

By induction assumption, each one of the disjuncts is equivalent in **SOL(Comp$_0$)** to an r-closed computational formula, so φ' is also equivalent to such a formula, by Lemma 7. □

3.2. Full second-order interpretation of arithmetic

The *full second-order interpretation of V_{FA}* has N as the formula that defines the target universe, with 0 interpreted by 0, s interpreted by s, and every other V_{FA}-identifier f interpreted by the formula G_f, in the usual sense. The following is a more detailed description of the latter point.

Let us say that an equation is *simple* if it is of the form $f(\vec{u}) = v$, where $\vec{u}, v$ are variables, and that a formula is *simple* if all equations therein are simple or are equations between atomic terms (i.e., variables or constants). For an equation E, let E^s be a simple formula, equivalent to E, obtained by hereditarily replacing equations by equivalent existential simple formulas. For example, $f(g(u)) = v$ is replaced by $\exists w\, (\, g(u) = w \wedge f(w) = v\,)$.

For a V_{FA}-formula φ the *interpretation φ^I of φ* arises from φ by replacing each equation E (except simple equations and equations between atomic terms) by E^s, then replacing each simple equation $f(\vec{u}) = v$ by $G_f[\vec{u}, v]$, then relativizing quantifiers to N. Note that in the standard V_{FA}-structure $\forall \vec{x}, z\, (\, G_f[\vec{x}, z] \leftrightarrow f(\vec{x}) = z\,)$, and so $\varphi \leftrightarrow \varphi^I$.

Given a second-order (or higher-order) formalism **C** (with constants 0 and s), the *fully-interpreted number theory of* **C**, **FNT[C]**, is

$$\begin{aligned} \mathbf{FNT[C]} \;&=_{\mathrm{df}}\; \{\varphi \mid \varphi \text{ is a closed } V_{FA}\text{-formula, and } \mathbf{C} \vdash \tilde{\varphi}^I\} \\ &=\; \{\varphi \mid \varphi \text{ is a closed } V_{FA}\text{-formula, and } \mathbf{C}, \neg(0 = 1) \vdash \varphi^I\}. \end{aligned}$$

3.3. Correctness of the full interpretation

We show that the full interpretation of V_{FA} in **SOL(Comp)**, $\varphi \mapsto \varphi^I$, is correct, that is, that the interpretation of each V_{FA}-identifier is the graph of a function over the interpreted universe N, provably in **SOL(Comp)**. We shall prove half of this already in **SOL(Comp$_0$)**, so our interpretation is "semi-correct" for **SOL(Comp$_0$)**.

LEMMA 15 *If f is a V_{FA}-identifier, then*

$$\mathbf{SOL(Comp_0)} \vdash \forall \vec{x}^N\, \exists z^N\, G_f[\vec{x}, z]$$

(where $arity(\vec{x}) = arity(f)$*).*

PROOF: By induction on $degree(f)$. The induction basis is trivial, and the case where f is defined by composition is straightforward.

Suppose f is defined by recurrence, $f(0,\vec{u}) = g(\vec{u})$; $f(\mathrm{s}(v),\vec{u}) = h(v,\vec{u},f(v,\vec{u}))$. By Lemma 14 the formula $\varphi[v] \equiv_{\mathrm{df}} N[v] \wedge \exists z^N G_f[v,\vec{y},z]$ is equivalent in **SOL(Comp$_0$)** to an r-closed computational formula, and so $N[x]$ implies, in **SOL(Comp$_0$)**,

$$(\forall v\, \varphi[v] \rightarrow \varphi[\mathrm{s}(v)]) \wedge \varphi[0] \;\rightarrow \varphi[x].$$

By induction assumption applied to G_g, $N[\vec{y}] \rightarrow \exists z^N G_g[\vec{y},z]$, and so $N[\vec{y}] \rightarrow \varphi[0]$, by the definition of G_f. By induction assumption applied to G_h we have

$$\forall w\,(N[v,\vec{y},w] \rightarrow \exists z^N G_h[v,\vec{y},w,z]),$$

so $N[\vec{y}]$ implies $\forall v\,(\varphi[v] \rightarrow \varphi[\mathrm{s}(v)])$. We have proved $N[x,\vec{y}] \rightarrow \varphi[x]$, which trivially implies the statement of the lemma. □

LEMMA 16 *If φ is a formula generated by propositional connectives and first-order quantifiers from atomic formulas, from formulas of the form $G_f[\vec{t}]$, and from formulas of the form $N[t]$, then comprehension for φ is provable in* **SOL(Comp)**.

PROOF: Analogous to the proof of Lemma 10, using Lemma 14 for the base case. □

LEMMA 17 *Let f be a V_{FA}-identifier. Then*

$$\mathbf{SOL(Comp)} \vdash \forall \vec{x}^N, z^N, s^N\,(G_f[\vec{x},z] \wedge G_f[\vec{x},s] \rightarrow z = s)$$

(where $arity(\vec{x}) = arity(f)$*).*

PROOF: By induction on $degree(f)$. The cases for initial functions and for definition by composition are straightforward.

If f is defined by recurrence, then $G_f[y,\vec{x},z]$ is of the form $\forall Q\,\chi$, where

$$\begin{aligned}\chi \;\equiv\; & (\forall \vec{u}, w\,(G_g[\vec{u},w] \rightarrow Q(0,\vec{u},w)) \\ & \wedge \forall v,\vec{u},w,a\,(Q(v,\vec{u},a) \wedge G_h[v,\vec{u},a,w] \rightarrow Q(\mathrm{s}(v),\vec{u},w))) \\ & \qquad \rightarrow Q(y,\vec{x},z).\end{aligned}$$

Let

$$\psi[p,\vec{q},r] \;\equiv_{\mathrm{df}}\; G_f[p,\vec{q},r] \wedge (N[p,\vec{q},r] \rightarrow \forall s\,(G_f[p,\vec{q},s] \rightarrow s = r)).$$

By Lemma 16 comprehension for ψ is provable in **SOL(Comp)**, so $G_f[y,\vec{x},z]$ implies $\chi[\psi/Q]$. The first conjunct of the premise of $\chi[\psi/Q]$ is provable in **SOL(Comp)**, by induction assumption applied to the function g. Using $N[p]$ and induction assumption applied to the function h, the second conjunct in the premise of $\chi[\psi/Q]$ is also provable. Thus, the conclusion is provable, i.e. we have proved, under the assumption $G_f[x,\vec{y},z]$, that

$$N[y,\vec{x},z] \rightarrow \forall s\,(\,G_f[x,\vec{y},s] \rightarrow s=z\,),$$

i.e. the statement of the lemma. □

From Lemmas 15 and 17 we have

PROPOSITION 18 *The full interpretation of V_{FA} in* **SOL(Comp)** *is correct; that is, for every V_{FA}-identifier f,*

$$\mathbf{SOL(Comp)} \;\vdash\; \forall\vec{x}^N\,\exists! y^N\,G_f[\vec{x},y].$$

3.4. Soundness of the full interpretation

We now show that the full interpretation of **FA** is sound for **SOL(Comp)**, and that the full interpretation of $\mathbf{\Sigma_1 A}$ is sound for $\mathbf{SOL(Comp_0)}$.

LEMMA 19 *If $E \in$ **PR**, then* $\mathbf{SOL(Comp_0)} \vdash N[var(E)] \rightarrow E^I$.

PROOF: Without loss of generality, let E be the principal equation of a definition by recurrence, $\forall x,\vec{y}\,(\,f(\mathrm{s}(x),\vec{y}) = h(x,\vec{y},f(x,\vec{y}))\,)$. Then

$$E^I \equiv\; \forall x^N,\vec{y}^N\,\exists u^N,v^N\,(\,G_f(x,\vec{y},u) \wedge G_h(x,\vec{y},u,v) \wedge G_f(\mathrm{s}(x),\vec{y},v)\,).$$

This is provable in $\mathbf{SOL(Comp_0)}$ by Lemma 15 and the definition of G_f. □

LEMMA 20 *Let f be a V_{FA}-identifier. If φ is an equality axiom of the form*

$$\varphi \;\equiv\; \forall x_1,\ldots,x_r,y\,(\,x_i=y \;\rightarrow\; f(x_1,\ldots,x_r)=f(x_1,\ldots,x_{i-1},y,x_{i+1},\ldots,x_r)\,),$$

then $\mathbf{SOL(Comp_0)} \vdash \varphi^I$.

PROOF: We have

$$\varphi^I \;\equiv\; \forall x_1^N,\ldots,x_r^N,y^N\,(\,x_i=y \\ \rightarrow \exists u^N\,G_f[x_1,\ldots,x_{i-1},y,x_{i+1},\ldots,x_r,u] \,\wedge\, G_f[x_1,\ldots,x_r,u]\,).$$

This is immediate by Lemma 15. □

THEOREM 21 *Let φ be a V_{FA}-formula. If* $\mathbf{FA} \vdash \varphi$, *then*

$$\mathbf{SOL(Comp)},\; \neg(0=1) \vdash N[var(\varphi)] \;\rightarrow\; \varphi^I.$$

and

$$\mathbf{SOL(Comp)} \vdash N[var(\varphi)] \;\rightarrow\; \tilde{\varphi}^I.$$

PROOF: The proof of the first half of the proposition is essentially the same as for Proposition 12. Different are only the cases for equality axioms, which follow here from Lemmas 19 and 20.

Analogously, $\mathbf{FA}^- \vdash \varphi$ implies $\mathbf{SOL(Comp)} \vdash N[var(\varphi)] \rightarrow \varphi^I$, which, combined with Lemma 1, yields the second half of the theorem. □

To prove an analogous statement for $\mathbf{\Sigma_1 A}$ and $\mathbf{SOL(Comp_0)}$ we first make the following observation.

LEMMA **22** *If φ is an existential V_{FA}-formula, then comprehension for φ^I is provable in* $\mathbf{SOL(Comp_0)}$.

PROOF: Immediate from Lemma 14. □

THEOREM **23** *Let φ be a V_{FA}-formula. If $\mathbf{\Sigma_1 A} \vdash \varphi$, then*

$$\mathbf{SOL(Comp_0)},\ \neg(0=1) \vdash N[var(\varphi)] \rightarrow \varphi^I.$$

and

$$\mathbf{SOL(Comp_0)} \vdash N[var(\varphi)] \rightarrow \tilde{\varphi}^I.$$

PROOF: Similar to the proof of Theorem 13, except that Lemma 22 is needed to justify the interpretation of existential instances of induction. □

4. Faithfulness of the interpretations

In this section we prove the *faithfulness*, i.e. completeness for $\mathbf{SOL(Comp)}$, of the interpretations of $\mathbf{FA}$, and the completeness of for $\mathbf{SOL(Comp_0)}$ of the interpretations of $\mathbf{\Sigma_1 A}$. To do so we need to consider Theorem 4 more formally. Using a surjective coding of sequences by numbers (à la Kleene [Kle52]), we let

$$Init[Q,x] =_{\mathrm{df}} \forall i{<}lth(x)\ (Q(i) \wedge (x)_i = 1)\ \vee\ (\neg Q(i) \wedge (x)_i = 0),$$

i.e., x is an initial segment of the characteristic function of Q. Put

$$\mathrm{WKL}_0 =_{\mathrm{df}} \forall R(\ (\forall Q\, \exists x\ Init[Q,x] \wedge R(x))\ \rightarrow\ \exists h\, \forall Q\, \exists x{<}h\ Init[Q,x] \wedge R(x)\,),$$

i.e., if R is a unary predicate over the universal binary tree, that contains an element on every branch, then there is a bound on the height of these elements.

Let $\mathbf{BT}$ be $\mathbf{SOL_0}$ with comprehension for quantifier-free formulas, quantifier-free induction, and $\mathbf{PR}$. This is the same as the Base Theory of [Sie87], but formulated with relational variables. Let Σ_1^0 denote, as usual, the set of existential (first-order) V_{FA}-formulas, and $\tilde{\Sigma}_1^0$ the set of existential V_{FA}-formulas with relational parameters. A formalization of the proof of Theorem 4 establishes the following.

LEMMA 24 *Every computational formula is equivalent, provably in* $\mathbf{BT} + \mathrm{WKL}_0$, *to a* $\tilde{\Sigma}_1^0$ *formula.*

Every r-closed computational formula is equivalent, provably in $\mathbf{BT} + \mathrm{WKL}_0$, *to a* Σ_1^0 *formula.* □

We shall need the following two technical lemmas. If Φ is a collection of formulas, then $\mathbf{Ind}(\Phi)$ denotes induction for all $\varphi \in \Phi$.

LEMMA 25 $\mathbf{BT} + \mathbf{SOL}(\tilde{\Sigma}_1^0) + \mathrm{WKL}_0 + \forall x N[x]$ *is conservative over* $\mathbf{FA}$.

PROOF: Suppose

$$\mathbf{BT},\ \mathbf{SOL}(\tilde{\Sigma}_1^0),\ \mathrm{WKL}_0,\ \forall x\, N[x]\ \vdash\ \varphi,$$

where φ is first-order. Then

$$\mathbf{BT},\ \mathbf{SOL}(\tilde{\Sigma}_1^0),\ \mathrm{WKL}_0,\ \mathbf{Ind}(\tilde{\Sigma}_1^0)\ \vdash\ \varphi.$$

However, the straightforward proof of WKL_0 (in fact of full König's Lemma) is easily derivable in $\mathbf{BT} + \mathbf{SOL}(\tilde{\Sigma}_1^0)$ (compare [Fri69, Theorem 3]). So we get

$$\mathbf{BT},\ \mathbf{SOL}(\tilde{\Sigma}_1^0),\ \mathbf{Ind}(\tilde{\Sigma}_1^0)\ \vdash\ \varphi.$$

The latter theory is well-known to be conservative over $\mathbf{FA}$ (see e.g. [Tro73, §1.9.4]), concluding the proof. □

Proving the analogous statement for $\mathbf{\Sigma_1 A}$ requires a little more:

LEMMA 26 $\mathbf{BT} + \mathbf{SOL}(\Sigma_1^0) + \mathrm{WKL}_0 + \forall x N[x]$ *is conservative over* $\mathbf{\Sigma_1 A}$.

PROOF: Suppose

$$\mathbf{BT},\ \mathbf{SOL}(\Sigma_1^0),\ \mathrm{WKL}_0,\ \forall x\, N[x]\ \vdash\ \varphi,$$

where φ is first-order. Then, as in the previous proof,

$$\mathbf{BT},\ \mathbf{SOL}(\Sigma_1^0),\ \mathrm{WKL}_0,\ \mathbf{Ind}(\Sigma_1^0)\ \vdash\ \varphi,$$

i.e.,

$$\mathbf{BT},\ \mathrm{WKL}_0,\ \mathbf{Ind}(\Sigma_1^0)\ \vdash\ \chi \to \varphi,$$

where χ is the conjunction of instances of Σ_1^0-comprehension. Sieg [Sie87] showed that the latter theory is conservative over $\mathbf{BT} + \mathbf{Ind}(\Sigma_1^0)$ with respect to Π_1^1-sentences. (The formulation WKL used in [Sie87] for Weak König's Lemma is easily seen to imply WKL_0 (in **BT**), so the result of [Sie87] applies to WKL_0.) Since χ is Σ^1_1, $\chi \rightarrow \varphi$ is Π^1_1, and so we get

$$\mathbf{BT},\ \mathbf{Ind}(\Sigma^0_1) \vdash \chi \rightarrow \varphi,$$

i.e.,

$$\mathbf{BT},\ \mathbf{SOL}(\Sigma^0_1),\ \mathbf{Ind}(\Sigma^0_1) \vdash \varphi.$$

Again, the latter theory is conservative over $\mathbf{\Sigma_1 A}$ (e.g. by the syntactic argument of [Tro73, §1.9.4]), yielding the lemma. □

LEMMA **27** $\mathbf{SOL_0}, \mathbf{PR} \vdash N[\mathrm{s}(x)] \rightarrow N[x]$.

PROOF: Let $\mathbf{p}$ denote the predecessor function, for which the defining equations are $\mathbf{p}(0) = 0$, $\mathbf{p}(\mathrm{s}(x)) = x$. By Lemma 6, $N[\mathrm{s}(x)]$ implies $N[\mathbf{p}(\mathrm{s}(x))]$, from which $N[x]$ by the definition of $\mathbf{p}$. □

LEMMA **28** *For every V_{FA}-identifier f,*

$$\mathbf{BT}, \mathbf{SOL}(\mathbf{Comp}) \vdash N[\vec{x}] \rightarrow (G_f[\vec{x}, z] \leftrightarrow f(\vec{x}) = z).$$

PROOF: By induction on *degree*(f). The cases for initial functions are trivial. If f is defined by composition, $f(\vec{x}) = h(g_1(\vec{x}), \ldots, g_k(\vec{x}))$, then, in $\mathbf{BT} + \mathbf{SOL}(\mathbf{Comp}) + N[\vec{x}]$,

$$\begin{array}{lll} G_f[\vec{x}, z] & \leftrightarrow\ \exists v_1 \ldots v_k.\ G_{g_1}(\vec{x}, v_1) \wedge \cdots \wedge G_{g_k}(\vec{x}, v_k) \wedge G_h(\vec{v}, z) & \text{by the definition of } G_f \\ & \leftrightarrow\ \exists v_1^N \ldots v_k^N.\ G_{g_1}(\vec{x}, v_1) \wedge \cdots \wedge G_{g_k}(\vec{x}, v_k) \wedge G_h(\vec{v}, z) & \text{by Lemma 18} \\ & \leftrightarrow\ \exists v_1^N \ldots v_k^N.\ g_1(\vec{x}) = v_1 \wedge \cdots \wedge g_k(\vec{x}) = v_k \wedge h(\vec{v}) = z & \text{by induction assumption} \\ & \leftrightarrow\ f(\vec{x}) = z \qquad \text{by the definition of } f \text{ and Lemma 18.} & \end{array}$$

Suppose that f is defined by recurrence: $f(0, \vec{u}) = g(\vec{u})$, $f(\mathrm{s}(v), \vec{u}) = h(v, \vec{u}, f(v, \vec{u}))$. Let $\varphi[v, \vec{u}, z] \equiv_{\mathrm{df}} (N[v, \vec{u}] \rightarrow f(v, \vec{u}) = z)$. Assume $G_f[a, \vec{x}, z]$. Then, by Lemma 16,

$$\begin{array}{ll} \forall \vec{u}, y\, (G_g[\vec{u}, y] \rightarrow \varphi[0, \vec{u}, y]) & \\ \quad \wedge \forall \vec{u}, v, w, y\, (\varphi[v, \vec{u}, w] \wedge G_h[v, \vec{u}, w, y] \rightarrow \varphi[\mathrm{s}(v), \vec{u}, y]) & (*) \\ \quad \rightarrow \varphi[a, \vec{x}, z]). & \end{array}$$

Note that $G_g[\vec{u}, y]$ and $N[\vec{u}]$ imply, by induction assumption, $g(\vec{u}) = y$, from which $f(0, \vec{u}) = y$. Thus $G_g[\vec{u}, y] \rightarrow \varphi[0, \vec{u}, y]$, i.e. the first premise of (*).

Towards proving the second premise of (*) assume $\varphi[v, \vec{u}, w] \wedge G_h[v, \vec{u}, w, y] \wedge N[\mathrm{s}(v), \vec{u}]$. Then, by Lemma 27, $N[v, \vec{u}]$, so $f(v, \vec{u}) = w$ by $\varphi[v, \vec{u}, w]$, and so also $N[w]$, by Lemma 18. From $G_h[v, \vec{u}, w, y]$ we then have, by induction assumption, $h(v, \vec{u}, w) = y$, and so $f(\mathrm{s}(v), \vec{u}) = y$. This establishes the second conjunct in (*). Thus, (*) implies $\varphi[a, \vec{x}, z]$, proving $G_f[a, \vec{x}, z] \rightarrow f(a, \vec{x}) = z$.

To prove the converse, assume $f(a, \vec{x}) = z$, and assume

$$\forall \vec{u}, y\,(\,G_g[\vec{u}, y] \rightarrow Q(0, \vec{u}, y)\,) \tag{5}$$

and

$$\forall \vec{u}, v, w, y\,(\,Q(v, \vec{u}, w) \wedge G_h[v, \vec{u}, w, y] \rightarrow Q(\mathrm{s}(v), \vec{u}, y)\,). \tag{6}$$

Let

$$\psi[v] \equiv_{\mathrm{df}} \forall y\,(\,N[v, \vec{u}] \wedge f(v, \vec{u}) = y \;\rightarrow Q(v, \vec{u}, y)\,).$$

We have $\psi[0]$ from (5), the induction assumption for g, and the first equation for f. Similarly, $\psi[v] \rightarrow \psi[\mathrm{s}(v)]$ follows from (6) and the induction assumption for h. Thus $N[a]$ implies that (5) and (6) imply $\psi[a]$. Therefore, $N[a, \vec{x}]$ and $f(a, \vec{x}) = z$ imply that (5) and (6) imply $Q(a, \vec{x}, z)$, i.e. $G_f[a, \vec{x}, z]$. This concludes the backward direction for the case of definition by recurrence, and the proof of the lemma. □

LEMMA **29** *For every V_{FA}-equation E, $E \leftrightarrow E^I$ is provable in* **BT** + **SOL**(Comp$_0$) + $\forall x.N[x]$.

PROOF: Straightforward, by Lemma 28. □

LEMMA **30** *For every V_{FA}-formula φ, $\varphi \leftrightarrow \varphi^N$ and $\varphi \leftrightarrow \varphi^I$ are provable in* **BT** + **SOL(Comp)** + $\forall x.N[x]$.

PROOF: By induction on φ. The basis is trivial for the first equivalent, and is established by Lemma 29 for the second equivalence. The induction step is trivial. □

THEOREM **31 DNT[SOL(Comp)] = FNT[SOL(Comp)] = FA.**

I.e., the following conditions are equivalent, for any closed V_{FA}-formula φ:

1. **FA** $\vdash \varphi$;
2. **SOL(Comp)**, **PR**, $\neg(0=1) \vdash \varphi^N$;
3. **SOL(Comp)**, **PR** $\vdash \tilde{\varphi}^N$;
4. **SOL(Comp)**, $\neg(0=1) \vdash \varphi^I$;
5. **SOL(Comp)** $\vdash \tilde{\varphi}^I$.

PROOF: The inclusions **FA** $\subseteq$ **DNT[SOL(Comp)]** and **FA** $\subseteq$ **FNT[SOL(Comp)]** are Theorems 13 and 21 above.

We prove the converse for the full interpretation (the case of the direct interpretation is identical). Suppose

$$\mathbf{SOL(Comp)} \vdash \tilde{\varphi}^I.$$

Then, by Lemma 24,

$$\mathbf{BT},\ \mathbf{SOL}(\tilde{\Sigma}_1^0),\ \mathrm{WKL}_0 \vdash \tilde{\varphi}^I,$$

and so, by Lemmas 30 and 1,

$$\mathbf{BT},\ \mathbf{SOL}(\tilde{\Sigma}_1^0),\ \mathrm{WKL}_0,\ \forall x N[x] \vdash \varphi.$$

By Lemma 25 this implies $\mathbf{FA} \vdash \varphi$. □

An analogous proof, using Lemma 26 in place of 25, yields:

THEOREM **32** $\mathbf{DNT[SOL(Comp_0)]} = \mathbf{FNT[SOL(Comp_0)]} = \mathbf{\Sigma_1 A}$.

I.e., the five conditions of Theorem 31 are equivalent, with $\mathbf{\Sigma_1 A}$ *in place of* **FA**, *and* $\mathbf{Comp_0}$ *in place of* **Comp**.

We summarize in the following theorems the semantic readings, based on Theorem 3, of Theorems 31 and 32. First, we have the following semantic characterizations of **FA**:

THEOREM **33** *Let φ be a closed V_{FA}-formula. The following conditions are equivalent:*

1. *φ is provable in* **FA**;
2. *$\tilde{\varphi}$ [respectively, φ] is valid, as a statement about N, in all Henkin-models of* **PR** *that are closed under computational definitions (i.e. abstract jump) [and in which 0 and 1 are distinct];*
3. *$\tilde{\varphi}$ [respectively, φ] is valid, as a statement about N, in all Henkin-structures that are closed under computational definitions [and in which 0 and 1 are distinct], and where the primitive recursive functions are defined by their graphs.*

Analogously, we have semantic characterizations of $\mathbf{\Sigma_1 A}$:

THEOREM **34** *Let φ be a closed V_{FA}-formula. The conditions in Theorem 33 are equivalent, with* $\mathbf{\Sigma_1 A}$ *in place of* **FA**, *and with "r-closed computational" in place of "computational".*

References

Bar69 Jon Barwise, *Applications of strict-Π^1_1 predicates to infinitary logic,* **Journal of Symbolic Logic 34** (1969) 409–423.

Bar75 Jon Barwise, **Admissible Sets and Structures,** Springer-Verlag, Berlin and New York, 1975.

Fag74 Ronald Fagin, *Generalized first order spectra and polynomial time recognizable sets,* in R. Karp (ed.). **Complexity of Computation,** SIAM-AMS, 1974, 43–73.

Fag75 Ronald Fagin, *Monadic generalized spectra,* **Zeitschrift für mathematische Logik und Grundlagen der Mathematik 21** (1975) 89–96.

Fri69 Harvey Friedman, *König's Lemma is weak,* Mimeographed note, Stanford University, 1969.

GV85 Haim Gaifman and Moshe Vardi, *A simple proof that connectivity of finite graphs is not first-order definable,* **Bulletin of the EATCS 26** (June 1985) 43–45.

Hen50 Leon Henkin, *Completeness in the theory of types,* **Journal of Symbolic Logic 15** (1950) 81–91.

Imm87 Neil Immerman, *Languages which capture complexity classes,* **SIAM Journal of Computing 16** (1987) 760–778.

JS74 N.G. Jones and A.L. Selman, *Turing machines and the spectra of first-order formulas,* **Journal of Symbolic Logic 39** (1974) 139–150.

Kle52 S.C. Kleene, **Introduction to Metamathematics**, Wolterns-Noordhof, Groningen, 1952.

KV87 Phokion Kolaitis and Moshe Vardi, *The decision problem for the probabilities of higher-order properties,* **Proceedings of the Nineteenth ACM Symposium on Theory of Computing,** ACM, Providence, 1987, 425–435.

KV88 Phokion Kolaitis and Moshe Vardi, *0-1 laws and the decision problem for fragments of Second-Order Logic,* **Proceedings of the Third IEEE Symposium on Logic in Computer Science,** IEEE, New York, 1988, 2–11.

Lei85 Daniel Leivant, *Logical and mathematical reasoning about programs,* **Conference Record of the Twelfth Annual Symposium on Principles of Programming Languages,** ACM, New York, 1985, 132–140.

Lei85a Daniel Leivant, *Hoare's logic captures program semantics,* Manuscript, July 1985.

Lei89 Daniel Leivant, *Descriptive characterizations of computational complexity,* **Journal of Computer and System Sciences 39** (1989) 51–83.

Lei90 Daniel Leivant, *Computationally based set existence principles,* in W. Sieg (ed.), **Logic and Computation,** Contemporary Mathematics, volume 106, American Mathematical Society, Providence, R.I., 1990, pp. 197–212.

Par77 Charles Parsons, *On a number-theoretic choice schema and its relation to induction,* in A. Kino et als. (eds.), **Intuitionism and Proof Theory**, North-Holland, Amsterdam, 1977, 459–473.

Pra65 Dag Prawitz, **Natural Deduction**, Almqvist and Wiskel, Uppsala, 1965.

Pre30 M. Presburger, *Über die Vollständigkeit eines gewissen Systems der Arithmetik ganzer Zahlen, in welchem die Addition als einzige Operation hervortritt,* **Sprawozdanie z I Kongresu Matematików Krajów Słowiańskich** (Comptes-rendus du I[er] Congrès des Mathématiciens des Pays Slaves), **Warszawa 1929**, Warsaw, 1930, pp. 92–101 and 395.

Sie87 Wilfried Sieg, *Provably recursive functionals of theories with König's Lemma,* **Rend. Sem. Mat. Univers. Politecn. Torino, Fasc. Specale: Logic and Computer Science (1986)**, 1987, 75–92.

Tro73 Anne S. Troelstra, **Metamathematical Investigation of Intuitionistic Arithmetic and Analysis**, Springer-Verlag (LNM #344), Berlin, 1973.