Some New Exceptions for the Semantic Tableaux Version of the Second Incompleteness Theorem

Dan E. Willard ⋆

SpringerVerlag LNCS # 2381, pp. 281-297 Tableaux-2002 Proceedings

Abstract. This article continues our study of axiom systems that can verify their own consistency and prove all Peano Arithmetic's Π_1 theorems. We will develop some new types of exceptions for the Semantic Tableaux Version of the Second Incompleteness Theorem.

1 Introduction

Gödel's Second Incompleteness Theorem states that sufficiently strong axiom systems are unable to formally verify their own consistency. Let us define an axiom system α to be **Self-Justifying** iff

i) one of α's theorems will assert α's consistency (using some reasonable definition of consistency),
ii) and the axiom system α is in fact consistent.

It is well known [5, 6, 14] that Kleene's Fixed Point Theorem implies every r.e. axiom system α can be easily extended into a broader system α^* which satisfies condition (i). Kleene's proposal [6] was essentially for the system α^* to contain all α's axioms plus the one added axiom sentence below.

$+$ There exists no proof of 0=1 from the union of α with *"this sentence"*.

Kleene noted that it was easy to apply the Fixed Point Theorem to formally encode a self-referencing statement, similar to the sentence above. The catch is that α^* can be inconsistent even while its added axiom formally asserts α^* 's consistency. For this reason, Kleene, Rogers and Jeroslow [5, 6, 14] each emphatically warned their readers that most axiom systems similar to α^* were useless on account of their inconsistency, *although they were technically well-defined.* This problem arises in both Gödel's paradigm (where α extends Peano Arithmetic), as well in many more general settings [1, 2, 4, 12, 16, 19, 25], where a Gödel-like diagnoalization argument can be constructed to show that the *very presence* of the axiom $+$ causes the system α^* to become inconsistent.

We have recently published four articles [20, 23–25] about generalizations of the Second Incompleteness Theorem and exceptions to it that exist for Semantic Tableaux deductive calculi. Let $A(x, y, z)$ and $M(x, y, z)$ denote $x + y = z$ and $x * y = z$, and let us say an axiom system α *recognizes* Addition and

⋆ Supported by NSF Grant CCR 99-02726. Email = dew@cs.albany.edu.

Multiplication as **Total Functions** iff it can prove $\forall x \forall y \exists z \; A(x,y,z)$ and $\forall x \forall y \exists z \; M(x,y,z)$. We showed in [23, 25] that all axiom systems recognizing Addition and Multiplication as total functions and containing one additional Π_1 axiom, called "V", are unable to recognize their self–consistency, under the paradigm of the Semantic Tableaux Deductive Calculi. On the other hand, our papers [20, 24] did show that systems can verify some forms of their Tableaux self-consistency while retaining an ability to prove at least analogs of Peano Arithmetic's Π_1 theorems, if they treat Addition as a total function and Multiplication as a "non-total" 3-way relation $M(x,y,z)$.

Our objective in this paper will be to establish crisper and stronger versions of the preceding results. This topic is mathematically complex because there are several plausible definitions D_1 , D_2 , D_3 , ... indicating that an axiom system α can verify its "Semantic Tableaux Consistency". These definitions are equivalent to each other under strong logics, but a weak axiom system α is typically unable to prove these definitions of consistency are equivalent to each other. Thus, the question naturally arises as to which definition of self–justification is one choosing to study ?

In general, it is desirable to use the weakest possible definition among the alternatives D_1 , D_2 , D_3 , ... when one is seeking to generalize the Second Incompleteness Theorem. On the other hand, the opposite is true when one seeks to develop "boundary–case exceptions" to the Second Incompleteness Theorem that introduce some new type of "Self–Justifying formalism". They become more wide–reaching when they use the strongest feasible D_i available.

A system α 's weakest possible definition of Tableaux–Self–Justification is that α can prove the non–existence of a Smullyan–like Semantic Tableaux proof p of the theorem $0 = 1$, in a context where p's proper axioms are drawn from the system α (itself). This was the formal definition of Self-Justification we used in our Tableaux–2000 conference paper [23], as well as in [25]. The preceding paragraph explained why weaker definitions of Self–Justification *are better,* when developing generalizations of the Second Incompleteness Theorem. Thus, we need not further improve the results from our prior generalizations [23, 25] of the Second Incompleteness Theorem because there is little room available for further improvement, at least in the context of axioms systems that extend Q+V and recognize the consistency of their Semantic Tableaux deductive calculi.

On the other hand, there are substantial open questions remaining about whether and how far our exceptions to the Second Incompleteness Theorem, involving for example the "IS(A)" formalism of [20, 24], can indeed be further improved. The reason the latter topic is much more open than the former is that *weaker is certainly not better than stronger* for it. This is because one would ideally like the boundary–case exceptions to the Second Incompleteness Theorem to use the the strongest possible definition of Self–Justification, among the available alternatives D_1 , D_2 , D_3 , ...

For example, our IS(A) system of [20, 24] could recognize the non–existence of any Smullyan–like Tableaux proof p of the theorem $0 = 1$, all of whose proper axioms are drawn from IS(A). However *for fully arbitrary sentence* Ψ , IS(A) was not able to prove the unavailability of two Smullyan–like Tableaux contradictory proofs for Ψ and $\neg \Psi$ from IS(A).

Essentially, this paper will seek to investigate how close we can come to establishing the above effect for extensions of IS(A). The next section will define the notion of a " Π_1^- " sentence that will help clarify this issue. Section 1.2 will then define a new axiom system, called IS-1(A), that recognizes that no pair of contradictory Semantic Tableaux proofs of the two sentences Ψ and $\neg \Psi$ from IS-1(A) can exist, whenever Ψ is a " Π_1^- " sentence. The Remark 5 (at the end of Section 1.2) will give a short 1-paragraph summary of a recent discovery by us, showing that the preceding boundary–case exceptions to the Second Incompleteness Theorem do not generalize from Π_1^- to Π_2^- sentences.

1.1. Notation and Statement of Main Theorem:

Some added notation is needed before we can state our main theorems. Say a function $F(a_1, a_2, ...a_j)$ satisfies the **Non-Growth** property iff F satisfies $F(a_1, a_2, ...a_j) \leq Maximum(a_1, a_2, ...a_j)$ for all values of $a_1, a_2, ...a_j$. Seven examples of non-growth functions are *Integer Subtraction* (where $x - y$ is defined to equal zero when $x \leq y$), *Integer Division* (where $x \div y$ is defined to equal x when $y = 0$, and we round down to the nearest integer), $Maximum(x, y)$, $Logarithm(x)$, $Predecessor(x) = \text{Max}(\, x-1\, ,\, 0\,)$, $Root(x, y) = \lceil\, x^{1/y}\, \rceil$ and $Count(x, j)$ designating the number of "1" bits among x's rightmost j bits. These functions are called the **Grounding Functions**.

We will use a slight variant of Logic's conventional notation when discussing grounding functions. A *term* will be defined to be a constant, variable or function symbol (whose input arguments are recursively defined terms). If t is a term then the quantifiers in the wffs $\forall\, v \leq t \;\; \Psi(v)$ and $\exists\, v \leq t \;\; \Psi(v)$ are called *bounded quantifiers*. If Φ is a formula that uses the Grounding primitives as its function symbols and the two relation symbols of " $=$ " and " $\leq$ ", then this formula will be called both " Π_0^-" and " Σ_0^-" whenever all its quantifiers are bounded. For $n \geq 1$, a formula Υ shall be called " Π_n^- " iff it is written in the form $\forall v_1 \; \forall v_2 \; ... \; \forall v_k \quad \Phi$, where Φ is " Σ_{n-1}^- ". Likewise, Υ is called " Σ_n^- " iff it is written in the form $\exists v_1 \; \exists v_2 \; ... \; \exists v_k \quad \Phi$, where Φ is " Π_{n-1}^- ".

Our definitions of Σ_n^- and Π_n^- formulae are the same as the conventional definitions of Σ_n and Π_n formulae, except that the Addition and Multiplication function symbols are replaced with Grounding primitives. Since Subtraction and Division can represent Addition and Multiplication as relations, every conventional Π_1 formula can be translated into an Π_1^- formula that is equivalent to it under sufficiently strong models of Arithmetic.

Using the preceding notation, we can provide a more succinct summary of our main result. Let us say an axiom system α has a **Level-N** understanding of its own Semantic Tableaux consistency iff it can recognize that there exists no Smullyan-like Semantic Tableaux proofs, using α's proper axioms, of both the theorems Ψ and $\neg \Psi$, for all Π_N^- sentences Ψ . Let us say α has a **primitive** understanding of its own Semantic Tableaux consistency iff it can recognize that there exists no Smullyan-like Semantic Tableaux proof, using α's proper axioms, of the theorem $0 = 1$. Our prior IS(A) system of [20, 24] technically had only a primitive understanding of its own Semantic Tableaux consistency. However, we could have easily improved this result to the Level-Zero. In this paper, we will show how a stronger version of IS(A), called IS-1(A), can have a Level-1 understanding of its own Semantic Tableaux consistency. This is a non–trivial improvement over our prior result because there exists no decision procedure for enumerating all true Π_1^- sentences.

1.2 Definition of IS-1(A):

Let A denote an axiom system that proves theorems about the Grounding Functions. Our new axiom system, called IS-1(A), will be a 4-part self-justifying axiom system. Its Group-Zero, 1 and 2 axiom schemes will be defined analogously to their counterparts in the IS(A) formalism of [20, 24] (except for some unimportant notational changes). However IS-1(A)'s Group-3 scheme will have a quite different definition than IS(A)'s counterpart. It will essentially assert that the IS-1(A) formalism is Level-1 consistent. The formal 4-part defintion of IS-1(A) is given below:

Group-Zero: Two of the Group-zero axioms will define two initial constants that correspond to the integers 0 and 1. The third Group-zero axiom will indicate that Addition is a total function. (It will thus provide a means to define integers larger than 1.) Since our Grounding Function formalism does not technically use an Addition function symbol, the axiom (1) below will view Addition as an operation that is the inverse of Subtraction:

$$\forall x \, \forall y \, \exists z \quad x \; = \; z \; - \; y \tag{1}$$

Group-1: This axiom group will consist of a finite set of Π_1^- sentences, denoted as F , which assures that the seven Grounding functions have their conventional properties with regards to the "=" and "<" predicates when they are given constants as inputs. By this we mean that for each grounding function G , k-tuple $(n_1, n_2, ...n_k)$ and added constant m , the union of axiom system F with (1)'s added axiom will be sufficient to prove whichever one of the three conditions of $G(n_1, n_2, ...n_k) = m$, $G(n_1, n_2, ...n_k) < m$ or $G(n_1, n_2, ...n_k) > m$ is true. (Any finite set of Π_1^- sentences F with this property may be used to define Group-1. Our prior published papers expressed no strong preference about which F was employed.)

Group-2: Let $\mathrm{Prf}_A^{\Phi}(x, y)$ denote a Σ_0^- formula indicating that y is a proof of the theorem Φ from the axiom system A . For each Π_1^- sentence Φ, the Group-2 schema will contain an axiom of the form (2). Thus, the Group-2 scheme shall trivially endow IS-1(A) with a capacity to verify all A's Π_1^- theorems.

$$\forall \, y \quad \{ \; \mathrm{Prf}_{\,A}^{\,\Phi} \; (\; y \;) \quad \supset \quad \Phi \;\; \} \tag{2}$$

Group-3: This group will consist of one Π_1^- sentence stating essentially:

∗ There exists no two Semantic Tableaux proofs from the **Union** of the Group-1 and 2 axioms with **this sentence looking at itself** of both some Π_1^- sentence and its negation.

In order to formally encode $*$ as a Π_1^- sentence, let $\text{Pair}(x, y)$ denote a Σ_0^- formula indicating x is the Gödel number of a Π_1^- sentence and y is x's negation. Also, let $\text{Prf}_{\text{IS-1}(A)}(a, b)$ denote a Σ_0^- formula that indicates that b is the Gödel number of a semantic tableaux proof of the theorem a from the axiom system IS-1(A). In this context, the formal Gödel encoding of the axiom sentence $*$ can be approximated as:

$$\forall\, x\, \forall\, y\, \forall\, p\, \forall\, q \quad \neg\ [\ \text{Pair}(x, y)\ \wedge\ \text{Prf}_{\text{IS-1}(A)}(x, p)\ \wedge\ \text{Prf}_{\text{IS-1}(A)}(y, q)\ ] \quad (3)$$

Remark 1. The full formal description of IS-1(A)'s Group-3 axiom is somewhat more complicated than the abbreviated descriptions of this axiom's structure, given either by the Sentence ∗ or by the analogous Equation (3). The main added complication arises because the Group-3 axiom declares the consistency of a formal set of axioms that includes "itself" (in the words of Sentence ∗.) The general notion of an axiom including formally "itself" when it refers to the consistency of an axiom schema goes back to Kleene's 1938 paper [6] (as the first paragraph of this article had already indicated). Kleene's abbreviated description is insufficient to establish that Equation (3) can be encoded precisely as a Π_1^- sentence. To do this, one needs techniques similar to Appendixes B through D from our article [24]. We will not repeat such a construction here.

Remark 2. One can easily become initially confused by IS-1(A)'s Group-3 axiom because IS-1(A) is *not automatically consistent* by virtue of the simple fact that its Group-3 axiom declares: *"I am consistent"*. Using the nomenclature from the opening paragraph of our article, the difficulty is that it is plausible that IS-1(A) could satisfy Part–i but not the *equally important* Part-ii of the definition of Self–Justification (as had happened with the example of α^*, in the first paragraph of this article). The next chapter will prove that this difficulty does not occur with IS-1(A).

Remark 3. We wish to reinforce the point made by the preceding paragraph, and graphically illustrate how some seemingly minor modifications in IS-1(A)'s formalism will result in the construction of an inconsistent system. Pudlák has proven that no consistent extension α of Robinson's axiom system Q can prove that all Hilbert-styled proofs employing α's axioms are assured to be free of inconsistencies. Solovay subsequently observed [16] it was possible to use methods of Nelson and Wilkie-Paris [9, 19] to incrementally refine this particular theorem of Pudlák's so that it will generalize for essentially all axiom systems that simultaneously recognize Successor as a total function and that retain a capacity to prove all Peano Arithmetic's Π_1^- theorems. Hence, IS-1(A) will automatically *become inconsistent* if its Group-3 axiom *is simply revised* so that "Prf" specifies a Hilbert rather than Semantic Tableaux variant of proof.

Remark 4. Similarly, our "Q+V" version of the Second Incompleteness Theorem from [23, 25] demonstrates that it is infeasible to modify the IS-1(A) axiom system so that it can simultaneously recognize Multiplication as a total function and retain a *logically valid* analog of Equation (3)'s Group-3 axiom. There is insufficient space to explain here why Multiplication's totality is central for effectuating the Semantic Tableaux version of the Second Incompleteness Theorem. However, the reader can find some very good and detailed intuitive explanations for this phenomena in either the passage spanning pages 328–331 in our article [20] or in Remark 4.5 of [24].

Remark 5. During the last three months while this article was being refereed, we developed a new version of the Second Incompleteness Theorem which states that there exists a Π_1^- sentence W , provable from the $\mathrm{I}\Sigma_0$ fragment of Peano Arithmetic, such that no consistent axiom system α can prove W , prove Addition is a total function and simultaneously recognize its own Level-2 Semantic Tableaux consistency. There is no space to insert the proof of this added theorem here, and we will display it elsewhere. It implies that when A has the strength of Peano Arithmetic, it is impossible to devise a modification of IS-1(A) that is consistent and whose Group-3 axiom precludes the possibility of there existing simultaneous Semantic Tableaux proofs of both an arbitrary Π_2^- sentence and its negation. This inability to generalize our results from Level-1 to Level-2 consistency makes the main theorem-proofs, presented in the next chapter, even the more interesting.

Remark 6. Let us say that a formula $\Upsilon(v)$ is an **Initialization Segment** *relative to* an axiom system α if α can formally prove:

$$\Upsilon(0) \;\text{ and }\; \forall v \; \{ \; \Upsilon(v) \;\supset\; \Upsilon(v+1) \; \} \tag{4}$$

Kriesel-Takeuti, Nelson, Pudlák, Visser and Wilkie-Paris [4, 8, 9, 12, 18, 19] have illustrated several examples of Initialization-Formulae $\Upsilon(v)$, where an axiom system α can prove its *Semantic Tableaux* consistency local to such Initialization Segments. Thus if " $\text{BadTableaux}_{\alpha}(y)$ " denotes that y is a semantic tableaux proof of the theorem 0=1 from α, then there are several known axiom systems α that can prove localized consistency statements similar to:

$$\forall y \quad \{ \quad \Upsilon(y) \quad \supset \quad \neg\, \text{BadTableaux}_{\,\alpha}\,(\; y \;) \quad \} \tag{5}$$

The earliest version of (5) was discovered by Kriesel-Takeuti [8] in the rather specialized context of a Second Order Logic generalization of the Cut-Free Sequent Calculus. Nelson [9] showed that Robinson's Arithmetic Q can prove a version of Equation (5) about itself. Pudlák [4, 12] proved a much more general theorem showing a similar effect was applicable to any finitely axiomatized sequential theory (and also allowing for Wilkie-Paris's notion [19] of a Herbrand-restricted-consistency). This literature is not exactly relevant to Self-Justifying axiom systems, because our axiom systems do not have their consistency statements localized by an analog of Equation (5)'s formula $\Upsilon(y)$. However, this literature is probably the closest analog to our results that has been explored by other researchers. We especially encourage the readers to examine Pudlák's work [4, 12] because it proves Equation (5)'s effect generalizes *for all* finitely axiomatized sequential theories, which is a quite *noteworthy phenomena !*

2 Proof of Main Theorem

Let $I(\bullet)$ denote a function that maps an initial axiom system A onto a second axiom system, denoted as $I(A)$. Let us call the mapping–formalism $I(\bullet)$ **Consistency-Preserving** iff I(A) is consistent whenever the union of the axiom system A with Section 1.2's Group-Zero and Group-1 axiom schemes is consistent. Our objective will be to prove that IS-1($\bullet$) is "Consistency-Preserving". In our discussion, a sentence Ψ will be called PRENEX* iff it is written in the form $Q_1\, x_1\; Q_2\, x_2 ...\; Q_n\, x_n\;\; \theta(x_1, x_2 ... x_n)$ where $\theta(x_1, x_2 ... x_n)$ is a Σ_0^- formula and Q_i denotes either the symbol $\forall$ or $\exists$.

Our definition of a semantic tableaux proof will be very similar to the definitions used in say Fitting's or Smullyan's textbooks [3, 15]. Define a ***Φ*-Based Candidate Tree** for the axiom system α to be a tree structure whose root corresponds to the sentence $\neg\,\Phi$ *rewritten in PRENEX* normal form* and whose all other nodes are either axioms of α or deductions from higher nodes of the tree. Let the notation " $\mathcal{A} \implies \mathcal{B}$ " indicate that $\mathcal{B}$ is a valid deduction when $\mathcal{A}$ is an ancestor of $\mathcal{B}$ in the candidate tree T. In this notation, the deduction rules allowed in a candidate tree are:

1. $\Upsilon \wedge \Gamma \implies \Upsilon$ and $\Upsilon \wedge \Gamma \implies \Gamma$.
2. $\neg\neg\,\Upsilon \implies \Upsilon$. Other valid Tableaux rules for the " $\neg$ " symbol include: $\neg(\Upsilon \vee \Gamma) \implies \neg\Upsilon \wedge \neg\Gamma$, $\neg(\Upsilon \supset \Gamma) \implies \Upsilon \wedge \neg\Gamma$, $\neg(\Upsilon \wedge \Gamma) \implies \neg\Upsilon \vee \neg\Gamma$, $\neg\,\exists v\, \Upsilon(v) \implies \forall v \neg\, \Upsilon(v)$ and $\neg\,\forall v\, \Upsilon(v) \implies \exists v\, \neg \Upsilon(v)$
3. A pair of sibling nodes Υ and Γ is allowed in a candidate tree when their ancestor is $\Upsilon \vee \Gamma$.
4. A pair of sibling nodes $\neg\Upsilon$ and Γ is allowed in a candidate tree when their ancestor is $\Upsilon \supset \Gamma$.
5. $\exists v\, \Upsilon(v) \implies \Upsilon(u)$ where u denotes a newly introduced "Parameter Symbol".
6. $\forall v\, \Upsilon(v) \implies \Upsilon(t)$ where t denotes a parameter term. The "Parameter Terms" here are built out of the Grounding Functions, whose inputs are any set of constant symbols $c_1, c_2, .., c_m$ and parameter symbols $u_1, u_2, .., u_n$, where each symbol u_i **was previously** introduced by an ancestor of the node storing the new deduction " $\Upsilon(t)$ ".

Define a particular leaf-to-root branch in a candidate tree T to be **Closed** iff it contains both some sentence Υ and its negation $\neg\,\Upsilon$. A **Semantic Tableaux** proof of Φ is defined to be a candidate tree whose root stores the sentence $\neg\Phi$ (written in PRENEX* normal form) and all of whose root-to-leaf branches are closed. The only distinction between our definition of a semantic tableaux proof and some other conventional definitions in [3, 15, 19] is that we require Φ's proof tree to have its root store $\neg\,\Phi$ *rewritten in PRENEX* normal form,* whereas some other conventional definitions do not have the PRENEX* requirement. All our theorems will also hold if we drop the PRENEX* requirement, but the notation in our main proofs will be greatly simplified if we begin with the assumption that the root has been normalized into PRENEX* form.

Let T denote a Φ-Based Candidate Tree, β denote a branch of T , L and M denote two fixed constants, and VAL($\bullet$) denote a function that maps each term s (from β) onto an integer VAL(s), subject to the following constraints:

A. Suppose u represents a new parameter symbol that is introduced at a depth level d along the branch β . Then its value will satisfy the constraint:

$$\text{Val}(u) \;\leq\; \text{Min}(\ M\ ,\ L \cdot 2^d\ ) \tag{6}$$

B. VAL($\overline{c_K}$) $=$ K when $\overline{c_K}$ corresponds to one of the two particular constants, c_0 or c_1, defined by IS-1(A)'s Group-zero axiom and representing the two numbers of zero and one.

C. VAL(s)'s definition will generalize in the natural manner for terms s that contain function symbols F , i.e. VAL($F(s_1, s_2)$) = F(VAL(s_1),VAL(s_2)).

Our next definition will require the added notation convention listed below.

> Let Ψ^M denote a sentence identical to Ψ except that all the *previously-unrestricted* universal quantifiers will have their ranges *redefined* in Ψ^M to correspond to the subset of non-negative integers $\leq M$. (Bounded universal quantifiers and both bounded and unbounded existential quantifiers in Ψ will not have their ranges changed under Ψ^M .)

Our next definition will use the fact that a Φ-Based Candidate Tree **IS NOT** a Semantic Tableaux proof of Φ when at least one of the branches of this tree fails to be closed. Let us say a branch β of such a Φ-Based Candidate Tree is **Conservative** iff there exists an ordered triple $(L, M,$VAL$)$ where β satisfies the preceding conditions A–C, plus the following additional fourth constraint below. (We will also call such a branch **(L,M)–Conservative** in the special case where the particular values for (L, M) are fixed and known in advance.)

D. All sentences Ψ appearing on the branch β will be sentences where Ψ^M is valid in the Standard Model of the Natural Numbers.

Before proceeding further, it would be useful to explain the significance of Conservative Branches. Lemma 1 will state that no "candidate tree" can be a "semantic tableaux proof" when it draws its proper axioms from IS-1(A) and *simultaneously contains* a Conservative Branch. This fact will enable us to develop a formal proof–by–contradiction that the IS-1($\bullet$) axiom mapping must be Consistency-Preserving, because otherwise the algorithm PROBE (which shall defined be in Section 2.2) will construct the particular type of Conservative Branch, whose existence will be *strictly forbidden* by Lemma 1.

Lemma 1. *. Let α denote an axiom system whose every axiom sentence is written in PRENEX* normal form (similar to IS-1(A)). Then none of candidate trees drawing their proper axioms from α can simultaneously contain a Conservative Branch and constitute a "semantic tableaux proof".*

Proof. Very trivial because all α's axioms, as well as any sentence stored in a proof-tree's root, are written in PRENEX* form. In such a context, it is simply impossible for a branch β in a proof tree p to be closed without β containing both some Σ_0 sentence Υ and its negation $\neg\Upsilon$. The latter cannot possibly occur in a Conservative branch, because Part-D of our definition of conservativeness would then require that Υ and its negation $\neg\Upsilon$ both be simultaneously valid sentences in the Standard Model of the Natural Numbers. □

We will now explain roughly how we will use Lemma 1 to devise a proof–by–contradiction that verifies that the IS-1($\bullet$) axiom mapping is Consistency–Preserving. Let $\omega(x, y, p, q)$ be the Σ_0^- formula that corresponds to the square bracket expression on the right-side of Equation (3). This means that IS-1(A)'s Group-3 axiom (defined by Equation (3)) can be simply rewritten as:

$$\forall\, x\; \forall\, y\; \forall\, p\; \forall\, q \quad \neg\;\; \omega(x, y, p, q) \tag{7}$$

For any second formula $\theta(x, y, p, q)$, let us examine the properties of the following second sentence:

$$\forall\, x\; \forall\, y\; \forall\, p\; \forall\, q \quad \{\quad \omega(x, y, p, q) \;\supset\; \theta(x, y, p, q) \quad\} \tag{8}$$

Let us call Equation (8) a **Vacuous Truth** iff it satisfies the following conditions:

a. Equation (8) is a logically valid statement.
b. Although it is a valid sentence, Equation (8) will actually not indicate what it may first appear to imply. This is because no tuple (x, y, p, q) will actually satisfy either $\omega(x, y, p, q)$ or $\theta(x, y, p, q)$.

It is well known that "vacuous truths" are often useful intermediate steps appearing in proofs-by-contradiction. For example, a proof-by-contradiction can verify Equation (7)'s assertion by employing Equation (8)'s "vacuous truth" and showing that no tuple (x, y, p, q) can satisfy $\theta(x, y, p, q)$.

Our formal proof of IS-1($\bullet$)'s Consistency–Preserving property will rest on a proof–by-contradiction of this type. Its invoked formal constraint sentence " $\forall x \forall y \forall p \forall q \;\; \neg \;\; \theta(x, y, p, q)$ " will turn out to be a fairly simple consequence of Lemma 1. Our objective will be to use this $\theta-$statement and (8) to establish (7). The only moderately difficult part of our proof will be because the full meaning of vacuous truths are always awkward to directly visualize, because of the inherently non–constructive nature of their statements. However, it will be ultimately easy to follow our proof, provided one remembers the inherently non–constructive nature of vacuous truths.

2.1 Structure of Main Proof

We begin by listing some notation that our proof will require:

1. Max(p, q) will denote the maximum of the two numbers p and q.

2. Top(P, Q) will be an abbreviation for the following formula:

$$\forall p \; \forall q \; \forall x \; \forall y \quad \{ \; \text{Max}(p,q) < \text{Max}(P,Q) \quad \supset \quad \neg \; \omega(x,y,p,q) \; \} \tag{9}$$

3. Check(X, Y, P, Q) will denote a Boolean formula. In particular, let us recall that the condition $\omega(X, Y, P, Q)$ indicates P is a proof from the axiom system IS-1(A) of a Π_1^- sentence, called X , and Q is the proof of the negation of this sentence, called Y . In order to simplify our notation, let us assume that X denotes the sentence " $\forall a \; \phi(a)$ " and Y denotes " $\exists a \;\; \neg \; \phi(a)$ ". Then the symbol Check(X, Y, P, Q) will yield a Boolean value of TRUE if and only if the following statement is valid in the Standard Model of the Natural Numbers:

$$\forall \; a \qquad [\; a \leq \frac{1}{2} \cdot \text{Max}(P,Q) \;] \quad \supset \quad \phi(a) \tag{10}$$

4. Constraint(t, β) is a formula indicating that t denotes the Gödel number of a Semantic Tableaux "Candidate Tree" and β is a conservative branch in t . (Note that Lemma 1 indicates that when Constraint(t, β) is satisfied, t **CANNOT** possibly represent a semantic tableaux proof. This fact will be used by our proofs–by–contradiction to justify Theorems 1 and 2.)

We will now use the preceding notation to outline the structure of the proof of our main theorem, asserting the consistency of the IS-1(A) formalism. Our proof can be viewed as having two parts. One part (appearing in Section 2.2) will establish the validity of the two statements (11) and (12), listed at the bottom of this paragraph. Both these statements are vacuous truths, characterized by the usual property that there exists no tuple (X, Y, P, Q) satisfying the square-bracket condition on the left side of their horn clause. As a result of their vacuous nature, some readers may have difficulty fully visualizing the meaning of (11) and (12), until Section 2.2's proof for them is examined. We suggest that the readers not feel particularly concerned to decipher their exact meaning at this current juncture. Rather, one should just treat (11) and (12) as simply purely formal mathematical objects. Our present goal will be to show how our main theorem, asserting the consistency of the IS-1(A) formalism, follows easily from these statements, when one uses a method of proof–by–contradiction. In conjunction with our formal proofs of (11) and (12), given later in Section 2.2, this analysis will establish that the IS-1(A) axiom system is, indeed, consistent.

$$\forall X \; \forall Y \; \forall P \; \forall Q \quad \{ \; [\; \text{Check}(X,Y,P,Q) \; \wedge \; \omega(X,Y,P,Q) \; \wedge \; \text{Top}(P,Q) \;] \\ \supset \quad \exists \beta \;\; \text{Constraint}(Q,\beta) \; \} \tag{11}$$

$$\forall X \; \forall Y \; \forall P \; \forall Q \quad \{ \; [\; \neg\text{Check}(X,Y,P,Q) \; \wedge \; \omega(X,Y,P,Q) \; \wedge \; \text{Top}(P,Q) \;] \\ \supset \quad \exists \beta \;\; \text{Constraint}(P,\beta) \; \} \tag{12}$$

We need one preliminary proposition before we can prove our main result.

Theorem 1. *The combination of Equations (11) and (12) immediately imply the validity of the following statement:*

$$\forall X \ \forall Y \ \forall P \ \forall Q \quad \{ \ \text{Top}(P,Q) \ \ \supset \ \ \neg\, \omega(X,Y,P,Q) \ \} \tag{13}$$

Proof. Our proof rests on separately examining the two cases where the tuple (X,Y,P,Q) does and does not satisfy the condition Check(X,Y,P,Q).

In the first case, we can infer from Equation (11) that the formal predicate condition " $\omega(X,Y,P,Q) \ \wedge \ \text{Top}(P,Q)$ " does imply the validity of

$$\exists\, \beta \ \ \text{Constraint}(Q, \beta) \ . \tag{14}$$

However, Lemma 1's formal statement, translated into our new notation, amounts to the assertion that it is impossible for Equation (14) and $\omega(X,Y,P,Q)$ to be simultaneously valid (see the footnote [1] for the formal details substantiating this point). Hence, our forced conclusion is that if the condition Check(X,Y,P,Q) is valid, then the identity " $\text{Top}(P,Q) \ \supset \ \neg\, \omega(X,Y,P,Q)$ " must automatically hold in this case.

We can use almost the identical technique to prove the validity of the identity " $\text{Top}(P,Q) \ \supset \ \neg\, \omega(X,Y,P,Q)$ " in the alternate case where the condition Check(X,Y,P,Q) is false. The only difference is that the second case will use Equation (12) rather than (11) to arrive at a similar proof–by–contradiction. (In particular, its proof will be the same as the preceding case, except that our counterpart of Equation (14)'s intermediate step will now be the observation that β satisfies Constraint(P, β) .) □

Theorem 2. *Suppose the union of the axiom system* A *with Section 1.2's "Group-1" axiom schema is a consistent system. Then IS-1*(A) *is also a consistent axiom system. (In our formal nomenclature, this amounts to stating that the axiom–mapping formalism IS-1(* $\bullet$ *) is "consistency–preserving".)*

Proof. It is easy to derive Theorem 2 from Theorem 1 by using the method of proof–by–contradiction. In particular, suppose that the theorem was false and IS-1(A) was inconsistent. Using our notation convention, there would then exist a tuple (x,y,p,q) satisfying $\omega(x,y,p,q)$. From such a tuple (x,y,p,q) , we can certainly find a second such tuple (X,Y,P,Q) , that also satisfies this $\omega-$condition, but additionally possesses the minimal possible value for MAX(P,Q) among all tuples satisfying $\omega(x,y,p,q)$. In our notation, this means that (X,Y,P,Q) will satisfy the following dual condition:

$$\omega(X,Y,P,Q) \quad \wedge \quad \text{Top}(P,Q) \tag{15}$$

[1] The basic reason for this inherent incompatibility is that the formula $\omega(X,Y,P,Q)$, by definition, implies Q is a semantic tableaux proof of the theorem Y . In this context, the formula Constraint(Q, β) flatly contradicts the preceding statement, since Lemma 1 indicates that the presence of Q's "conservative branch" β demonstrates that it is impossible for the "candidate" tree Q to be a formal tableaux-style proof.

However, the point is that Theorem 1 precludes the possibility that any tuple could satisfy Equation (15) (because the latter blatantly contradicts the invariant (13), established by Theorem 1). Hence, our proof–by–contradiction forces us to conclude that the Theorem 2 must be valid, because otherwise the formal statement of Theorem 1 would be contradicted. □

The remainder of this chapter will justify the vacuous truths, from Equations (11) and (12), so that our proofs for Theorem 1 and 2 shall be formally completed.

2.2 Proofs of Equations (11) and (12)

To prove Equations (11) and (12), we need to first introduce some notation. Let T denote a candidate tree, and (L, M) denote the parameters used to define an $(L, M)-$Conservative Branch. The symbol PROBE will denote an algorithm which given these inputs, seeks to construct a valuation VAL($\bullet$) and a $(L, M)-$Conservative Branch for T, called Beta(T, L, M). The four algorithmic rules for constructing Beta(T, L, M) and VAL($\bullet$) are listed below:

1. The top node along the path Beta(T, L, M) will always be T's root.
2. Suppose the first i nodes along the path Beta(T, L, M) are $N_1, N_2, N_3, \ldots N_i$, and the node has N_i has two children denoted as N_a and N_b. In this case, the candidate tree T has used either the $\vee-$Elimination or $\supset-$Elimination to justify this binary split, and we will let Ψ_a and Ψ_b denote the two sentences stored in these two nodes. In this case, our algorithm PROBE(T, L, M) will make the "left child" N_a constitute the next element along Beta(T, L, M)'s path when the sentence Ψ_a^M is valid (in the Standard Model of the Natural Numbers). Otherwise, it will make N_b be Beta(T, L, M)'s next node.
3. If the first i nodes along the Beta(T, L, M)'s path are $N_1, N_2, N_3, \ldots N_i$ and N_i has only one child, denoted as N_{i+1}, then the algorithm PROBE(T, L, M) will "attempt" to make N_{i+1} the next node along Beta(T, L, M)'s path. This "attempt" may not be successful. The difficulty arises when N_{i+1}'s sentence is constructed via the $\exists-$Elimination Rule (and it thus introduces a new parameter–symbol, called say u_j). In this case, the procedure PROBE(T, L, M) will attempt to assign VAL(u_j) the smallest possible value (consistent with the assignments it previously gave to $u_1, u_2, \ldots u_{j-1}$). If the resulting quantity VAL(u_j) is sufficiently small to satisfy Equation (6)'s inequality, then PROBE(T, L, M)'s attempt will be considered successful. Otherwise, the procedure PROBE(T, L, M) will simply "quit" and cease attempting to build T's $(L, M)-$conservative branch, called Beta(T, L, M).
4. The procedure PROBE(T, L, M) will iteratively repeat Steps 2 and 3 to make the path Beta(T, L, M) become longer and longer, until either it reaches the candidate tree's desired leaf–level or a failure occurs in Step 3.

Our next two lemmas will show how we may apply the procedure PROBE(T, L, M) to corroborate the validity of Equations (11) and (12).

Lemma 2. *. Suppose the union of the axiom system* A *with IS-1*(A)*'s Group-zero and Group-1 axioms is a consistent axiom system and that the 4-tuple* (X,Y,P,Q) *satisfies the square-bracket expression on the left side of Equation (11). (This expression is rewritten below.)*

$$Check(X,Y,P,Q) \;\wedge\; \omega(X,Y,P,Q) \;\wedge\; Top(P,Q) \tag{16}$$

Let us set $L = 1$, $M = \frac{1}{2} \cdot Max(P,Q) - 1$ *and* $T = Q$ *. Then for these input values for* (T,L,M)*, the procedure PROBE will successfully find an* $(L,M)-$*Conservative Branch lying in the candidate tree* Q *.*

Lemma 3. *. Let us again suppose that the union of the axiom system* A *with IS-1*(A)*'s Group-zero and Group-1 axioms is a consistent axiom system. Also, suppose that the* (X,Y,P,Q) *satisfies the square-bracket expression on the left side of Equation (12). (This expression is rewritten below.)*

$$(\;\neg\; Check(X,Y,P,Q)\;) \;\wedge\; \omega(X,Y,P,Q) \;\wedge\; Top(P,Q) \tag{17}$$

Let us set $L = \frac{1}{2} \cdot Max(P,Q)$, $M = Max(P,Q) - 1$ *and* $T = P$ *. Then for these input values for* (T,L,M)*, the procedure PROBE will successfully find an* $(L,M)-$*Conservative Branch lying in the candidate tree* P *.*

PROOF OF LEMMA 2. We will use the Principle of Induction to prove Lemma 2. Our inductive proof shall assume that the i highest nodes $N_1, N_2...N_i$ along the path Beta(T,L,M) satisfy the $(L,M)-$Conservative condition, and it will use this fact to deduce that the node N_{i+1} will also be $(L,M)-$Conservative. Our inductive proof will be divided into eight sub–cases because we must separately consider the possibilities that N_{i+1} constitutes T's root, stores an axiom of IS-1(A), or is deduced from a higher node of the candidate tree T via one of the six Elimination rules for the $\exists$, $\forall$, $\wedge$, $\vee$, $\supset$ or $\neg$ symbols.

1. The Case where N_{i+1} **designates** T**'s root:** We will employ the notation from Equation (10)'s definition of Check(X,Y,P,Q). It indicated that if for some Σ_0^- formula $\phi(a)$, X denotes the sentence " $\forall a \; \phi(a)$ " and Y denotes " $\exists a \;\; \neg\; \phi(a)$ ", then Check(X,Y,P,Q) denotes the statement:

$$\forall\; a \;\; [\; a \leq \frac{1}{2} \cdot \text{Max}(P,Q)\;] \;\;\supset\;\; \phi(a) \tag{18}$$

In a context where Lemma 2 sets $M = \frac{1}{2} \cdot$ Max$(P,Q) - 1$ and $[\;\neg Y\;]$ denotes $\neg Y$ rewritten in Prenex* Normal form, Equation (18) implies that the sentence $[\;\neg\, Y\;]^M$ is valid. Moreover since Q represents a proof of the sentence Y , the root of Q's proof tree is the sentence " $\neg\, Y$ ". Hence the last two sentences show that the root satisfies Part-D of the definition of $(L,M)-$Conservativeness. (We do not need to verify it also satisfies the other three parts of this definition because the root contains no parameter symbols u , and thus these conditions hold trivially, by default.) □

2. The Case where N_{i+1} **stores one of IS-1**(A)**'s formal axioms.** Let Ψ denote the axiom sentence stored in N_{i+1} . Similar to the preceding case, the only non–trivial aspect of this case is the demonstration that Ψ satisfies Part-D of the definition of $(L, M)-$Conservativeness (i.e. that Ψ^M is valid under the Standard Model). The proof of this fact is divided into three sub–cases:

1. *Sub-case where Ψ is one of IS-1(A)'s Group-zero or Group-1 axioms:* In this case Ψ is automatically valid under the Standard Model of the Natural Numbers, and hence so is Ψ^M .
2. *Sub-case where Ψ is one of IS-1(A)'s Group-2 axioms:* Lemma 2's hypothesis indicates that the union of A with IS-1(A)'s Group-zero and Group-1 schemes is a consistent system. This fact implies that every Group-2 axiom is valid under the Standard Model. Moreover, Equation (2)'s definition of a Group-2 axiom indicates that these axioms are encoded as Π_1^- sentences. Such Π_1^- sentences Ψ have the property that Ψ's validity automatically implies the validity of Ψ^M .
3. *Sub-case where Ψ is IS-1(A)'s Group-3 axioms (formally defined by Equation (3)):* Unlike the other sub-cases, we shall not assume the validity of the sentence Ψ at the start of the proof of this sub-case (see footnote [2]). However, Lemma 2's hypothesis does indicate validity of $Top(P, Q)$ (i.e. see Equation (16)). The latter, combined with M's definition, immediately implies that Ψ^M is valid when Ψ denotes IS-1(A)'s Group-3 axiom. □

3. The Case where N_{i+1} is generated by the ∃-Elimination Rule: This Elimination rule, defined in Section 2's second paragraph, allows N_{i+1} to represent a sentence $\phi(u^*)$, containing a new parameter symbol u^* , when an ancestor of N_{i+1} represents the sentence $\exists v \; \phi(v)$. A key aspect of IS-1(A) is that Equation (1) is its only axiom using unbounded existential quantifiers. Let MaxVal(i) denote the maximum of 1 and of the largest quantity, Val(u), stored in the nodes $N_1, N_2, N_3, ...N_i$. It is clear that the elimination of an existential quantifier, originating from Equation (1), will cause Val$(u^*) \leq 2 \cdot$ MaxVal(i).

Moreover since neither any of IS-1(A)'s other proper axioms nor its root contains unbounded existential quantifiers, the preceding inequality clearly implies MaxVal$(i + 1) \leq 2 \cdot$ MaxVal(i). This latter inequality, combined with the facts that MaxVal(1)= 1 and the height of Q's proof tree is certainly less than $\frac{1}{3}$ Log$_2(M)$ immediately shows that the Val(u^*) will satisfy Equation (6)'s constraint. The proof that the node N_{i+1} will satisfy the other parts of the definition of (L, M)-Conservativeness is trivial. □

4. The Case where N_{i+1} is generated by the ∀-Elimination Rule: This Elimination rule, allows N_{i+1} to represent a sentence $\phi(u^*)$, when an ancestor of N_{i+1} stores the sentence $\forall v \; \phi(v)$ and u^* represents a parameter symbol used in one of N_{i+1}'s ancestors, one of the constants of 0 or 1, or a term generated from these primitive objects. In each of these cases, a routine inductive argument shows that $\phi(u^*)$ must satisfy the (L, M)-Conservative Condition

[2] It turns out that IS-1(A)'s Group-3 axiom is a valid statement. However, we cannot assume its validity during the course of Lemma 2's proof because Lemma 2's purpose is to help prove Theorem 2, and the validity of IS-1(A)'s Group-3 axiom is not evident until Theorem 2 is formally proven.

because the parameter symbols appearing inside u^* satisfied Equation (6) and because the higher node storing " $\forall v \ \phi(v)$ " should be inductively presumed to be (L,M)-Conservative. □

5. The Case where N_{i+1} is generated by the $\wedge$-Elimination Rule: Trivial because a sentence Υ will automatically satisfy the (L,M)-Conservative Condition when some ancestor of it storing the sentence $\Upsilon \wedge \Theta$ does. □

6. The Case where N_{i+1} is generated by a $\neg$ Elimination Rule: It is, once again, trivial that a sentence Υ will automatically satisfy the (L,M)-Conservative Condition when some ancestor of it storing the sentence $\neg\neg\Upsilon$ does. Also, a similar trivial argument applies to the other variants of $\neg$ Elimination (formally defined in Section 2's second paragraph). □

7. The Case where N_{i+1} is generated by the $\vee$-Elimination Rule: This rule introduces a pair of sibling nodes Υ and Θ when they have a common ancestor $\Upsilon \vee \Theta$. Since the inductive hypothesis implies $\Upsilon \vee \Theta$ satisfies the (L,M)-Conservative Condition, one of Υ or Θ must also satisfy this condition. Our algorithm PROBE will automatically select this satisfying node. □

8. The Case where N_{i+1} is generated by the $\supset$-Elimination Rule: Essentially the same as the preceding Case 7. □

PROOF OF LEMMA 3. The general structure of Lemma 3's proof will be analogous to Lemma 2's proof, in that it will again demonstrate the node N_{i+1} satisfies the $(L,M)-$Conservative condition with an inductive argument that presumes that the i higher nodes $N_1, N_2 ... N_i$ along the path Beta(T,L,M) already satisfy the $(L,M)-$Conservative condition. Our inductive proof will be divided into eight cases, six of which are the same as their analogs in Lemma 2's proof (i.e. Cases 2 and 4–8). The remaining two cases appear below:

A) The Case where N_{i+1} designates T's root: This case differs from the Case 1 of Lemma 2's proof because a proof of P of X stores $\neg X$ (which corresponds to Y rewritten in Prenex* Normal form) in its root, rather than $\neg Y$ (which corresponds to X in Prenex* Normal form). Another distinction is that Lemma 3's hypothesis assumes the validity of the condition $\neg$ Check(X,Y,P,Q), whereas Lemma 2's hypothesis presumed Check(X,Y,P,Q). The salient point is that this shift in Check(X,Y,P,Q)'s Boolean value allows us to conclude that the new sentence stored in T's root, under Lemma 3, is also valid in the Standard Model of the Natural Numbers. Hence if Ψ denotes the root's sentence, we can again conclude the Ψ^M is valid, showing that the root satisfies Part-D of the definition of $(L,M)-$Conservativeness. (Once again, it is trivial that the root satisfies the other three parts of the definition of $(L,M)-$Conservativeness.) □

B) The Case where N_{i+1}'s Stored Sentence is generated by the $\exists$-Elimination Rule: The reason this case is different from the Case 3 of Lemma 2's proof is that the root of T's candidate tree will, for some Π_0^- formula $\phi(v)$, correspond to a sentence of the form " $\exists \ v \ \phi(v)$ " in the current case. Moreover from the fact that Lemma 3's hypothesis indicates that the condition $\neg$ Check(X,Y,P,Q) is valid, we can presume our valuation will assure that Val$(u) \leq L$, whenever N_{i+1}'s stored sentence " $\phi(u)$ " is deduced by eliminating the existential quantifier from " $\exists \ v \ \phi(v)$ " .

The remainder of our proof for the current case is analogous to Case 3 from Lemma 2's proof. It uses again the fact that the only proper axiom of IS-1(A) containing an unbounded existential quantifier is Equation (1)'s axiom. In particular, let MaxVal(i) again denote the maximum of 1 and of the largest quantity, Val(u), stored in the nodes $N_1, N_2, N_3, ...N_i$. We already noted in Lemma 2's proof that $\text{MaxVal}(i+1) \leq 2 \cdot \text{MaxVal}(i)$, whenever the node N_{i+1} introduces a new parameter u, generated by eliminating an existential quantifier stemming from Equation (1). Hence, this observation, combined with the inequality from the preceding paragraph, implies $\text{MaxVal}(i+1) \leq \text{Max} \; [\; L \; , 2 \cdot \text{MaxVal}(i) \;]$.

This recurrence relation, together with the facts that $L = \frac{1}{2} \cdot \text{Max}(P, Q)$, $M = \text{Max}(P, Q) - 1$ and that P's proof tree has height less than $\frac{1}{3} \; \text{Log}_2(M)$, demonstrates that the parameter u, generated by the $\exists$-Elimination Rule meets Equation (6)'s requirements. It is, again, trivial to justify it satisfies the other parts of the definition of (L, M)-Conservativeness. □

The preceding proofs of Lemmas 2 and 3 also complete our justification for Theorems 1 and 2. This is because Section 2.1's proof of these two theorems had pre-supposed the correctness of Equations (11) and (12), an assumption which Lemmas 2 and 3 do now corroborate.

Generalizations of Theorem 2 and Added Perspectives. Let H denote a list of ordered pairs (t_1, p_1) , $(t_2, p_2)...(t_n, p_n)$, where p_k is a Semantic Tableaux proof of the theorem t_k . Define H to be a **R(i , j) Tableaux-Hierarchy Proof** of the theorem T from the axiom system α iff $T = t_n$ and H also satisfies the following two conditions:

1. The formal axioms used in p_m's proof are either one of $t_1, t_2, ...t_{m-1}$ or come from α.
2. Each of the sentences $t_1, t_2, ...t_{n-1}$ are required to have a Π_i^* or Σ_j^* format.

Consider a revised form of the IS-1(A) that uses $R(1,1)$ deduction rather than conventional semantic tableaux as its underlying formalism. Thus, this version of IS-1(A), which perhaps should be called IS-1*(A), will have an identical definition as IS-1(A) except that its Group-3 axiom will employ a variant of Eq (3) where "Prf" now denotes a $R(1,1)$ proof rather than a conventional tableaux proof. Thus IS-1*(A)'s Group-3 axiom will be identical to IS-1(A)'s counterpart except that it will have this type of Prf $_{\text{IS-1}^*(A)}$ predicate replace Prf $_{\text{IS-1}(A)}$.

A longer version of this paper generalizes Theorem 2 to establish that the IS-1*(•) axiom–mapping formalism is "consistency–preserving". Moreover, all our formalisms can also be further strengthened so that they support the additional properties of our article [24]'s Tangibility Reflection Principle.

It also turns out that Theorem 2 and its generalizations collapse when $R(2,1)$ deduction replaces $R(1,1)$. In this case, a generalization of the Second Incompleteness Theorem can establish there exists a Π_1^- sentence W , provable from the $I\Sigma_0$ fragment of Peano Arithmetic, such that no consistent axiom system α can prove W , prove Addition is a total function and simultaneously recognize the assured non-existence of a $R(2,1)$ proof of 0=1 using α's axioms.

References

1. Z. Adamowicz, "Herbrand Consistency and Bounded Arithmetic", *Fundamenta Mathematica* 171 (2002) pp. 279-292.
2. S. Buss, *Bounded Arithmetic,* Proof Theory Lecture Notes #3, Bibliopolic 1986.
3. M. Fitting, *First Order Logic & Automated Theorem Proving,* SpringerVerlag 1990.
4. P. Hájek and P. Pudlák, *Metamathematics of First Order Arithmetic,* Springer Verlag 1991.
5. R. Jeroslow, "Consistency Statements in Formal Mathematics", *Fundamenta Mathematica* 72 (1971) pp. 17-40.
6. S. Kleene, "On the Notation of Ordinal Numbers", *Journal of Symbolic Logic* 3 (1938), pp. 150-156.
7. J. Krajícek, *Bounded Propositional Logic and Complexity Theory,* Cambridge University Press, 1995.
8. G. Kreisel and G. Takeuti, "Formally Self-Referential Propositions for Cut-Free Classical Analysis", Dissertationes Mathematicae 118 (1974) pp. 1–55
9. E. Nelson, *Predicative Arithmetic,* Math Notes, Princeton Press, 1986.
10. R. Parikh, "Existence and Feasibility in Arithmetic", *Journal of Symbolic Logic* 36 (1971), pp.494-508.
11. J. Paris and C. Dimitracopoulos, "A Note on the Undefinability of Cuts", *Journal of Symbolic Logic* 48 (1983) pp. 564-569.
12. P. Pudlák, "Cuts, Consistency Statements and Interpretations", *Journal of Symbolic Logic* 50 (1985) pp.423-442.
13. P. Pudlák, "On the Lengths of Proofs of Consistency", in *Collegium Logicum: Annals of the Kurt Gödel Society Volume 2,* published (1996) by Springer-Wien-NewYork, pp 65-86.
14. H. Rogers, *Recursive Functions and Effective Compatibility,* McGrawHill 1967.
15. R. Smullyan, *First Order Logic,* Springer Verlag, 1968.
16. R. Solovay, Private Communications (1994) about his generalization of one of Pudlák's theorems from [12]. Solovay never published any of his observations about "Definable Cuts" that several logicians [4, 7, 9, 11–13, 19] have attributed to his private communications. We provide our short 4-page interpretation of Solovay's revised form of Pudlák's Incompleteness Theorem in Appendix A of [24].
17. G. Takeuti, *Proof Theory,* Studies in Logic Volume 81, North Holland, 1987.
18. A. Visser, "An Inside View of Exp", *Journal of Symbolic Logic* 57 (1992) 131–165
19. A. Wilkie and J. Paris, "On the Scheme of Induction for Bounded Arithmetic", *Annals of Pure and Applied Logic* (35) 1987, 261-302
20. D. Willard, "Self-Verifying Axiom Systems", *Third Kurt Gödel Colloquium* (1993), Springer-Verlag LNCS#713, pp. 325-336.
21. D. Willard, "The Tangibility Reflection Principle", *Fifth Kurt Gödel Colloquium* (1997), Springer-Verlag LNCS#1289, pp. 319–334.
22. D. Willard, "Self-Reflection Principles and NP-Hardness", *Dimacs Series # 39* (1998), pp. 297–320 (AMS Press).
23. D. Willard, "The Semantic Tableaux Version of the Second Incompleteness Theorem Extends Almost to Robinson's Arithmetic Q", in *Semantic Tableaux 2000 Conference Proceedings,* SpringerVerlag LNAI#1847, pp. 415-430.
24. D. Willard, "Self-Verifying Systems, the Incompleteness Theorem and the Tangibility Reflection Principle", in *Journal of Symbolic Logic* 66 (2001) pp. 536-596.
25. D. Willard, "How to Extend The Semantic Tableaux And Cut-Free Versions of the Second Incompleteness Theorem Almost to Robinson's Arithmetic Q", *Journal of Symbolic Logic* 67 (2002) pp. 465-496.