Exact Witness Architectures: Selection Jump and Limits of Exact Certification

Parker Emmerson[1*] and Ryan J. Buchanan[2]

[1*]Independent Researcher, Chapel Hill, NC, USA.
[2]Oregon State University, Corvallis, Oregon, USA.

*Corresponding author(s). E-mail(s): parkeremmerson@icloud.com;
Contributing authors: theryestbread@gmail.com;

Abstract

We study exact witness architectures, sentences or semantic classes equipped with distinguished exact witness channels. The main theorem is a selection jump: for every decidable local verifier, any nonempty stagewise-local success class is automatically $\mathbf{\Pi^0_2}$-complete. Thus a single successful seed already forces maximal stagewise complexity. Around this theorem we prove three further barrier layers. Finite stagewise prefixes are uniformly insufficient, and undecidable co-c.e. witness architectures admit no decidable one-shot positive certifier and no exact-domain compiler. Same-theory adequacy along a universal $\mathbf{\Pi_1}$ embedding yields full $\mathbf{\Pi_1}$ reflection, ruling out internal exact certification in consistent recursively axiomatizable extensions of $\boldsymbol{I\Sigma_1}$. Finally, an arithmetic exact terminality predicate exists on a truth-faithfully embedded fragment exactly when the fragment truth set is arithmetical, yielding Tarski and diagonal barriers. For fixed propositions with exact two-sided decidable witness packages, these results assemble into a bridge trichotomy: isolated extensional bridges are vacuous, effective bridge classes are empty or $\mathbf{\Pi^0_2}$-complete, and assertion-enriched resolver layers are truth-universal.

Keywords: exact witness architecture, selection jump, exact certification, theorem proving, arithmetic truth, diagonalization

MSC Classification: 03D20 , 03D32 , 03F25 , 03F40 , 68T15

1 Introduction

This paper studies a representation-sensitive metamathematics of exact recognition. Many mathematical statements admit equivalent formulations, but once some of those formulations are singled out as the *exact witness channels* through which closure or success is to be recognized, the resulting architecture imposes strong computability-theoretic, proof-theoretic, and definability-theoretic constraints. We call such a structure an *exact witness architecture*. From the standpoint of automated reasoning, the framework formalizes the distinction between finite progress certificates, exact terminal certifiers, and internal certification procedures.

Two witness shapes recur throughout the paper:

$$\forall w\, \neg B(w) \qquad \text{and} \qquad \forall n\, \exists c\, R(n,c),$$

corresponding respectively to bad-witness exclusion and stagewise positive certification. Our first observation is that finite approximation is often not terminal. For stagewise channels, no fixed finite prefix is a uniform bridge across all decidable relations R, even though for each fixed represented R the universal closure of all prefixes is already equivalent in Robinson arithmetic to the full stagewise statement. We then prove a second lower-layer obstruction: if

$$A = \{x : \forall w\, \neg B(x,w)\}$$

is an undecidable co-c.e. witness architecture, then negative instances have finite bad witnesses, but positive instances admit neither a decidable one-shot positive certifier nor a partial computable exact-domain compiler.

The paper's main structural theorem concerns *stagewise-local witness classes*. Given a decidable local verifier $\mathsf{Cert}(n,c)$, define

$$S_{\mathsf{Cert}} = \left\{ e : \forall n\, \exists t\, \big(\mathsf{T}_{\mathrm{K}}(e,n,t) \wedge \mathsf{Cert}(n, \mathsf{U}_{\mathrm{K}}(t))\big) \right\}.$$

An element of S_{Cert} is a *successful seed*. The central theorem shows that nonemptiness already forces full Π^0_2 universality.

Selection Jump Theorem. *For every decidable* $\mathsf{Cert}(n,c)$*, if* $S_{\mathsf{Cert}} \neq \varnothing$*, then* S_{Cert} *is* Π^0_2*-complete.*

This theorem isolates a structural phenomenon: stagewise locality is not a mild form of closure. Once one successful seed exists, the whole success class jumps to maximal complexity at the Π^0_2 level. The source of this jump is distinct from both truth-theoretic and proof-theoretic barriers; it already appears for purely arithmetic exact predicates and does not depend on an internal truth predicate or a background provability predicate.

Above the selection jump lie two higher barrier layers. The first is *reflection collapse*. We construct a universal Π_1 witness class by coding Π_1 truth uniformly into non-halting, and show that same-theory adequacy along a Π_1-universal embedding yields full Π_1 reflection. Hence no consistent recursively axiomatizable extension of

$I\Sigma_1$ can internally certify exact success at that level. The second is *definability collapse*. We prove an exact threshold theorem: an arithmetic exact terminality predicate exists on a truth-faithfully embedded fragment if and only if the fragment truth set is arithmetical. This yields a witness-architecture form of Tarski's barrier, and together with diagonal universality yields inconsistency for full internal biconditional schemes.

These results assemble into a synthesis for fixed propositions with exact two-sided decidable witness packages. The naive notion of a single bridge sentence is extensionally vacuous. The meaningful bridge object is instead the effective bridge class, which is empty if the proposition is false and Π^0_2-complete if it is true. If one enlarges further to a resolver layer that allows either positive stagewise certification or negative bad-witness output, and then enriches that layer by arbitrary arithmetic assertions, one reaches a truth-universal class. This is the bridge trichotomy proved in the paper. A benchmark example shows that the resulting barriers are structural rather than disguised restatements of bare difficulty: the same architecture already appears over a trivially provable proposition.

Position relative to existing work.

The lower-layer asymmetry results are close in spirit to classical c.e./co-c.e. barriers and index-set phenomena, while the reflection and truth-theoretic layers draw on standard Gödel and Tarski obstructions [1, 2, 4–7]. The point of the present paper is not to reprove those results in isolation, but to identify a single architectural setting in which they arise uniformly. In particular, the Selection Jump Theorem is not a theorem about one pre-chosen Π^0_2 set: it states that every nonempty stagewise-local class with decidable local verification is forced to be Π^0_2-complete. Likewise, the Exact Threshold Theorem is a characterization result: arithmetic exactness on a truth-faithful image is possible precisely up to the arithmetical boundary of the corresponding fragment truth set.

Relevance to automated reasoning.

From the standpoint of automated reasoning, a witness architecture formalizes the distinction between finite progress certificates, exact terminal certifiers, and internal certification procedures. Stagewise-local classes model proof-producing or solver-producing procedures that must return locally checkable certificates on every query or stage; exact-domain compilers model procedures that halt exactly on successful instances; and same-theory adequacy captures the ambition that a proof system or verification environment internally recognize exactly those procedures that are correct. The barrier theorems therefore identify structural limits on exact certification, staged verification, and self-verifying proof environments.

Section 2 introduces exact witness architectures and the basic exhaustion principle for exact channels. Section 3 treats finite prefixes and positive-certification asymmetry. Section 4 proves the Selection Jump Theorem. Section 5 establishes reflection collapse for universal Π_1 embeddings. Section 6 proves the exact threshold theorem and derives the Tarski and diagonal barriers. Section 7 gives the bridge trichotomy for fixed propositions. Section 8 provides a benchmark calibration. Technical coding lemmas are collected in Appendix A.

This paper is purely metamathematical. Concrete theorem-level witness packages for specific mathematical statements may be studied within the framework developed here, but they are not needed for the abstract barrier theory itself.

2 Exact Witness Architectures and Preliminaries

Convention 2.1 (Computability conventions) Fix once and for all:

(i) an acceptable numbering $(\varphi_e)_{e\in\mathbb{N}}$ of the partial computable functions;

(ii) primitive recursive Kleene normal-form predicates $\mathsf{T}_{\mathrm{K}}(e,n,t)$ and output extraction $\mathsf{U}_{\mathrm{K}}(t)$;

(iii) a standard primitive recursive coding of finite tuples by natural numbers;

(iv) a standard primitive recursive Gödel coding of arithmetic formulas and sentences;

(v) a standard acceptable numbering $(W_e)_{e\in\mathbb{N}}$ of the c.e. subsets of $\mathbb{N}$.

Write

$$\mathsf{TOT} := \{e : \forall n\, \exists t\, \mathsf{T}_{\mathrm{K}}(e,n,t)\}.$$

Also define

$$\mathsf{K}_0 := \{e : \exists t\, \mathsf{T}_{\mathrm{K}}(e,0,t)\}.$$

Convention 2.2 (Representations of primitive recursive maps and predicates) For every primitive recursive function $f : \mathbb{N}^k \to \mathbb{N}$ fixed in a theorem or proof, choose once and for all an arithmetic formula $F_f(\vec{x}, z)$ representing its graph, with the numeral-correctness property that for every $\vec{n} \in \mathbb{N}^k$,

$$\mathsf{Q} \vdash \forall z \left(F_f(\bar{\vec{n}}, z) \leftrightarrow z = \overline{f(\vec{n})}\right).$$

Similarly, for every fixed primitive recursive predicate $R(\vec{x})$ that appears inside arithmetic formulas, we write $R(\vec{x})$ also for a chosen arithmetic formula whose numeral instances are correctly decided by Q:

$$R(\vec{n}) \implies \mathsf{Q} \vdash R(\bar{\vec{n}}), \qquad \neg R(\vec{n}) \implies \mathsf{Q} \vdash \neg R(\bar{\vec{n}}).$$

Thus expressions such as $\overline{f(n)}$ are meta-level abbreviations for numerals, while quantified uses of primitive recursive maps are to be read via their fixed graph formulas F_f.

Convention 2.3 (Reducibility convention) Unless explicitly stated otherwise, completeness claims for subsets of $\mathbb{N}$ are with respect to primitive recursive many-one reducibility.

Convention 2.4 (Restricted-domain primitive recursive maps) Whenever a map is said to be primitive recursive on a syntactic domain (for example, on sentence codes or formula codes), we mean that it is the restriction of a total primitive recursive function on $\mathbb{N}$.

Convention 2.5 Fix a first-order language L and a background theory S in that language. All derivability statements in this section are ordinary first-order derivability over S. When later sections pass to arithmetic theories extending S, we shall say so explicitly.

Definition 2.6 (Exact Witness Architecture) Let P be an L-sentence. An *exact witness architecture* of P over S is a finite or recursively presented family of L-sentences

$$\mathsf{Arch}(P) = \{\sigma_i(P) : i \in I\}$$

such that

$$S \vdash P \leftrightarrow \sigma_i(P) \qquad \text{for every } i \in I.$$

The $\sigma_i(P)$ are the *exact witness channels* of P over S.

Remark 2.7 An exact witness architecture is stronger than a mere list of equivalent reformulations. The exact witness channels are the distinguished exact routes through which success is to be recognized relative to the chosen background theory.

Example 2.8 (Common witness-channel shapes) Two common exact witness-channel shapes are:

(i) a *bad-witness channel*

$$\forall w\, \neg B(w);$$

(ii) a *stagewise certificate channel*

$$\forall n\, \exists c\, R(n, c).$$

Proposition 2.9 (Channel Exhaustion) *Let P have an exact witness architecture over S, and let $T \supseteq S$ be any theory in the same language. Then for every exact witness channel $\sigma_i(P)$,*

$$T \vdash P \qquad \Longleftrightarrow \qquad T \vdash \sigma_i(P).$$

In particular, if T proves one exact witness channel, then T proves every exact witness channel.

Proof Since $S \vdash P \leftrightarrow \sigma_i(P)$, monotonicity gives

$$T \vdash P \leftrightarrow \sigma_i(P).$$

Therefore $T \vdash P$ if and only if $T \vdash \sigma_i(P)$. □

Corollary 2.10 (Bridge Necessity) *Let P have an exact witness architecture over S. If $T \supseteq S$ proves P, then T proves every exact witness channel of P. Equivalently, if T proves no exact witness channel, then T does not prove P.*

Proof Immediate from Proposition 2.9. □

Definition 2.11 (Bridge) Let P have an exact witness architecture over S. A sentence B is a *bridge* for P over S if

$$S \vdash B \rightarrow \sigma_i(P)$$

for one, equivalently every, exact witness channel $\sigma_i(P)$. By Proposition 2.9, this is equivalent to $S \vdash B \rightarrow P$.

3 Finite Approximation and Positive-Certification Asymmetry

3.1 Finite prefixes and ω-closure

Definition 3.1 (Stagewise channel and finite prefixes) Let $R(n,c)$ be a decidable relation on $\mathbb{N}^2$. Define

$$\sigma_R :\Longleftrightarrow \forall n\, \exists c\, R(n,c).$$

For $M \in \mathbb{N}$, define

$$\mathsf{Pref}_R(M) :\Longleftrightarrow \forall n < M\, \exists c\, R(n,c).$$

Theorem 3.2 (Uniform Finite-Prefix Insufficiency) *For every $M \in \mathbb{N}$ there exists a primitive recursive relation $R_M(n,c)$ such that*

$$\mathbb{N} \models \mathsf{Pref}_{R_M}(M) \qquad \textit{but} \qquad \mathbb{N} \models \neg\sigma_{R_M}.$$

Consequently, there is no theorem schema valid uniformly for all decidable stagewise predicates R that infers σ_R from the single finite prefix $\mathsf{Pref}_R(M)$.

Proof Fix M, and define

$$R_M(n,c) \;:\Longleftrightarrow\; (n < M \wedge c = 0).$$

Then R_M is primitive recursive. For each $n < M$, choosing $c = 0$ witnesses $R_M(n,c)$, so

$$\mathbb{N} \models \mathsf{Pref}_{R_M}(M).$$

But no c satisfies $R_M(M,c)$, hence

$$\mathbb{N} \models \neg\sigma_{R_M}.$$

□

Proposition 3.3 (Universal Prefix Closure Is Already Finitarily Equivalent) *Let $R(n,c)$ be a represented arithmetic relation. Then*

$$\mathsf{Q} \vdash \forall M\Big(\forall n < M\, \exists c\, R(n,c)\Big) \leftrightarrow \forall n\, \exists c\, R(n,c).$$

Consequently,

$$\mathbb{N} \models \sigma_R \qquad \Longleftrightarrow \qquad \mathbb{N} \models \forall M\, \mathsf{Pref}_R(M).$$

Proof For the forward implication, Q proves

$$\forall n\, \exists c\, R(n,c) \to \forall M\, \forall n < M\, \exists c\, R(n,c)$$

by universal instantiation.

For the reverse implication, assume

$$\forall M\Big(\forall n < M\, \exists c\, R(n,c)\Big).$$

Fix n. Instantiating at $M = S(n)$, we obtain

$$\forall m < S(n)\, \exists c\, R(m,c).$$

Since $\mathsf{Q} \vdash n < S(n)$, it follows that $\exists c\, R(n,c)$. As n was arbitrary,

$$\mathsf{Q} \vdash \forall M\Big(\forall n < M\, \exists c\, R(n,c)\Big) \to \forall n\, \exists c\, R(n,c).$$

The final equivalence in $\mathbb{N}$ is immediate. □

Definition 3.4 (ω-logic over Q) Let Q^ω denote Robinson arithmetic Q augmented with the ω-rule

$$\frac{\varphi(\overline{0}) \qquad \varphi(\overline{1}) \qquad \varphi(\overline{2}) \qquad \cdots}{\forall x\, \varphi(x)}.$$

Theorem 3.5 (ω-Closure of Numeral Prefix Theoremhood) *Let $R(n,c)$ be a decidable relation represented in arithmetic. Suppose that for every $M \in \mathbb{N}$,*

$$\mathsf{Q} \vdash \mathsf{Pref}_R(\overline{M}).$$

Then

$$\mathsf{Q}^\omega \vdash \sigma_R.$$

Proof By the ω-rule,

$$\mathsf{Q}^\omega \vdash \forall M\, \mathsf{Pref}_R(M).$$

By Proposition 3.3,

$$\mathsf{Q} \vdash \forall M\, \mathsf{Pref}_R(M) \leftrightarrow \sigma_R.$$

Hence

$$\mathsf{Q}^\omega \vdash \sigma_R.$$

□

3.2 Undecidable co-c.e. witness architectures

Definition 3.6 (co-c.e. witness architecture) Let $B(x,w)$ be a decidable relation on $\mathbb{N}^2$. Define

$$A_B := \{x \in \mathbb{N} : \forall w\, \neg B(x,w)\}.$$

Thus positive membership in A_B means the absence of bad witnesses.

Definition 3.7 (Decidable one-shot positive certifier) Let $A \subseteq \mathbb{N}$. A decidable relation $C(x,c)$ is a *decidable one-shot positive certifier* for A if

$$x \in A \qquad \Longleftrightarrow \qquad \exists c\, C(x,c).$$

Theorem 3.8 (No One-Shot Positive Certifier for Undecidable co-c.e. Witness Architectures) *Let $B(x,w)$ be decidable, and let*

$$A_B = \{x : \forall w\, \neg B(x,w)\}.$$

If A_B is undecidable, then there is no decidable one-shot positive certifier for A_B.

Proof Because B is decidable,

$$x \notin A_B \iff \exists w\, B(x,w),$$

so A_B is co-c.e. Suppose there were a decidable $C(x,c)$ with

$$x \in A_B \iff \exists c\, C(x,c).$$

Then A_B would be c.e. as well. Therefore A_B would be both c.e. and co-c.e., hence decidable. Contradiction. □

Definition 3.9 (Exact-domain witness compiler) Let $A \subseteq \mathbb{N}$, and let $W(x, c)$ be decidable. A partial computable function f is an *exact-domain witness compiler* for A through W if

$$x \in A \qquad \Longleftrightarrow \qquad f(x)\downarrow \, \wedge \, W(x, f(x)).$$

Theorem 3.10 (No Exact-Domain Witness Compiler on Undecidable co-c.e. Families) *Let $A \subseteq \mathbb{N}$ be co-c.e. and undecidable, and let $W(x, c)$ be decidable. Then there is no exact-domain witness compiler for A through W.*

Proof Suppose f were such a compiler, with index e. Then

$$x \in A \iff \exists t \left(\mathsf{T}_{\mathrm{K}}(e, x, t) \wedge W(x, \mathsf{U}_{\mathrm{K}}(t))\right).$$

The right-hand side is c.e. because T_{K} and W are decidable. Hence A would be c.e. Since A is also co-c.e., it would be decidable. Contradiction. □

Corollary 3.11 (Positive-Negative Asymmetry) *Let $A_B = \{x : \forall w \, \neg B(x, w)\}$ with B decidable and A_B undecidable. Then:*

(i) negative instances have finite bad witnesses:

$$x \notin A_B \iff \exists w \, B(x, w);$$

(ii) positive instances admit no decidable one-shot positive certifier;

(iii) positive instances admit no partial computable exact-domain witness compiler.

Proof Item (i) is immediate from the definition. Item (ii) is Theorem 3.8. Item (iii) is Theorem 3.10. □

4 Stagewise Witness Classes and the Selection Jump

4.1 Stagewise families and effective bridge realizers

Definition 4.1 (Stagewise witness family) Let $R(x, n, c)$ be a decidable relation on $\mathbb{N}^3$. Define

$$A_R := \{x \in \mathbb{N} : \forall n \, \exists c \, R(x, n, c)\}.$$

Theorem 4.2 (Universal Sweep Reduction) *For every decidable relation $R(x, n, c)$,*

$$A_R \leq_m \mathsf{TOT}$$

via a primitive recursive many-one reduction. Hence every stagewise witness family A_R belongs to Π^0_2.

Proof For each x, define a partial computable function $\psi_x(n)$ by searching for the least c such that $R(x, n, c)$, and outputting it when found. Since R is decidable, ψ_x is partial computable uniformly in x. By the *s-m-n* theorem there is a primitive recursive map f with $\varphi_{f(x)} = \psi_x$. Then

$$x \in A_R \iff \forall n \, \exists c \, R(x, n, c) \iff \forall n \, \psi_x(n)\downarrow \iff f(x) \in \mathsf{TOT}.$$

Therefore $A_R \leq_m \mathsf{TOT}$ via the primitive recursive reduction f, and the displayed definition gives $A_R \in \Pi^0_2$. □

Definition 4.3 (Effective bridge realizer) Let $R(x, n, c)$ be decidable. An index e is an *effective bridge realizer* for x (relative to R) if

$$\forall n\, \exists t\, \big(\mathsf{T}_{\mathrm{K}}(e, n, t) \wedge R(x, n, \mathsf{U}_{\mathrm{K}}(t))\big).$$

Definition 4.4 (Bridge existence set) For decidable $R(x, n, c)$, define

$$\mathsf{BridgeExists}_R := \Big\{x : \exists e\ \forall n\, \exists t\, \big(\mathsf{T}_{\mathrm{K}}(e, n, t) \wedge R(x, n, \mathsf{U}_{\mathrm{K}}(t))\big)\Big\}.$$

Theorem 4.5 (Bridge Existence Collapse) *For every decidable $R(x, n, c)$,*

$$\mathsf{BridgeExists}_R = A_R.$$

Equivalently, for every x,

$$\exists e\ \forall n\, \exists t\, \big(\mathsf{T}_{\mathrm{K}}(e, n, t) \wedge R(x, n, \mathsf{U}_{\mathrm{K}}(t))\big) \iff \forall n\, \exists c\, R(x, n, c).$$

Proof The left-to-right implication is immediate.

Conversely, assume $\forall n\, \exists c\, R(x, n, c)$. Define the search machine $\psi_x(n)$ that outputs the least such c. Since R is decidable and the assumption guarantees existence for each n, ψ_x is total. Let e_x be an index of ψ_x. Then for every n, there exists a computation code t such that

$$\mathsf{T}_{\mathrm{K}}(e_x, n, t) \wedge R(x, n, \mathsf{U}_{\mathrm{K}}(t)).$$

So e_x is an effective bridge realizer for x, and $x \in \mathsf{BridgeExists}_R$. □

4.2 Stagewise-local witness classes

Definition 4.6 (Stagewise-local witness class) Let $\mathsf{Cert}(n, c)$ be a decidable predicate on $\mathbb{N}^2$. Define

$$\mathsf{Suc}_{\mathsf{Cert}}(e) \ :\iff\ \forall n\, \exists t\, \big(\mathsf{T}_{\mathrm{K}}(e, n, t) \wedge \mathsf{Cert}(n, \mathsf{U}_{\mathrm{K}}(t))\big).$$

Write

$$S_{\mathsf{Cert}} := \{e : \mathsf{Suc}_{\mathsf{Cert}}(e)\}.$$

A code e_0 with $\mathsf{Suc}_{\mathsf{Cert}}(e_0)$ is a *successful seed.*

Proposition 4.7 (Stagewise-Local Success Is Π^0_2) *For every decidable $\mathsf{Cert}(n, c)$, the success set S_{Cert} is Π^0_2.*

Proof Immediate from the definition. □

Theorem 4.8 (Selection Jump Theorem) *Assume $\mathsf{Cert}(n, c)$ is decidable and e_0 is a successful seed. Then for every decidable predicate $R(a, n, m)$ there exists a primitive recursive injective map*

$$f_R : \mathbb{N} \to \mathbb{N}$$

such that for every $a \in \mathbb{N}$,

$$\mathsf{Suc}_{\mathsf{Cert}}(f_R(a)) \iff \forall n\, \exists m\, R(a, n, m).$$

Proof Fix e_0 with $\mathsf{Succ}_{\mathsf{Cert}}(e_0)$. For each a, define a partial computable machine M_a by the following procedure on input n:

(i) search for an m such that $R(a, n, m)$;

(ii) if such an m is found, simulate $\varphi_{e_0}(n)$;

(iii) when $\varphi_{e_0}(n)$ halts with output c, output c;

(iv) if no such m exists, diverge.

Because R is decidable, this is partial computable uniformly in a. By the *s-m-n* theorem and padding, there is a primitive recursive injective map f_R assigning to a an index of M_a.

If $\forall n \, \exists m \, R(a, n, m)$, then the first search succeeds for every n, and because e_0 is a successful seed, $\varphi_{e_0}(n)$ outputs a valid certificate on every input n. Hence $\mathsf{Succ}_{\mathsf{Cert}}(f_R(a))$.

Conversely, if $\exists n_0 \, \forall m \, \neg R(a, n_0, m)$, then $f_R(a)$ diverges at input n_0, so $\neg\mathsf{Succ}_{\mathsf{Cert}}(f_R(a))$. □

Remark 4.9 (Why the selection jump is not merely a restatement of TOT-hardness) The theorem is not about one particular stagewise set A_R. It says that every class of the stagewise-local form

$$\forall n \, \exists t \, \big(\mathsf{T}_{\mathrm{K}}(e, n, t) \wedge \mathsf{Cert}(n, \mathsf{U}_{\mathrm{K}}(t))\big)$$

with decidable local verification obeys a dichotomy: it is either empty or Π^0_2-complete, and this conclusion is triggered solely by the existence of a single successful seed. The content is therefore structural, not a one-off reduction from TOT to a preselected class.

Corollary 4.10 (No Intermediate Nonempty Stagewise-Local Class) *For every decidable* $\mathsf{Cert}(n, c)$*, exactly one of the following holds:*

(i) $S_{\mathsf{Cert}} = \varnothing$*;*

(ii) S_{Cert} *is* Π^0_2*-complete.*

Proof By Proposition 4.7, $S_{\mathsf{Cert}} \in \Pi^0_2$. If $S_{\mathsf{Cert}} \neq \varnothing$, choose a successful seed and apply Theorem 4.8 to

$$R(a, n, m) :\Longleftrightarrow \mathsf{T}_{\mathrm{K}}(a, n, m).$$

Then

$$a \in \mathsf{TOT} \iff f_R(a) \in S_{\mathsf{Cert}}.$$

So $\mathsf{TOT} \leq_m S_{\mathsf{Cert}}$. Since TOT is Π^0_2-complete by Proposition A.1, S_{Cert} is Π^0_2-complete. □

Example 4.11 (A source-distinct instance of the selection jump) Let

$$\mathsf{Cert}_0(n, c) :\Longleftrightarrow (c = 0).$$

Then

$$\mathsf{Succ}_{\mathsf{Cert}_0}(e) \iff \forall n \, \exists t \, \big(\mathsf{T}_{\mathrm{K}}(e, n, t) \wedge \mathsf{U}_{\mathrm{K}}(t) = 0\big),$$

which is itself an arithmetic Π_2 predicate. The constant-zero total function is a successful seed, so by Theorem 4.8 and Corollary 4.10, the class S_{Cert_0} is Π^0_2-complete. This example shows that the source of the jump is not Tarski undefinability and does not require a same-theory provability predicate.

5 Universal Π_1 Witness Classes and Reflection Collapse

The coding lemmas used in this section are collected in Appendix A.

Theorem 5.1 (Uniform Π_1-Coding by Non-Halting) *There exists a primitive recursive predicate $U(e,t)$ with the following property.*

For every bounded arithmetic formula $\delta(v,w)$ whose free variables are among v,w, there exists a primitive recursive injective function

$$s_\delta : \mathbb{N} \to \mathbb{N}$$

such that for every $x \in \mathbb{N}$,

$$I\Sigma_1 \vdash \forall w\, \delta(\bar{x},w) \leftrightarrow \forall t\, U(\overline{s_\delta(x)},t).$$

In particular, for every Π_1-sentence φ, there exists $e \in \mathbb{N}$ such that

$$I\Sigma_1 \vdash \varphi \leftrightarrow \forall t\, U(\bar{e},t).$$

Proof Fix primitive recursive projection functions π_0, π_1 for the pairing map, and define

$$U(e,t) \;:\Longleftrightarrow\; \neg \mathsf{T}_{\mathrm{K}}(\pi_0(e),\pi_1(e),t).$$

Then U is primitive recursive.

Now let $\delta(v,w)$ be bounded, and let $d := \ulcorner \delta(v,w) \urcorner$. Apply Lemma A.3 to the primitive recursive predicate

$$R_\delta(x,w) \;:\Longleftrightarrow\; \neg \mathsf{Sat}_{\Delta_0}(d,x,w).$$

Let $p_\delta := s_{R_\delta}$, and define

$$s_\delta(x) := \langle p_\delta(x), 0 \rangle.$$

Fix $x \in \mathbb{N}$. By Lemma A.3,

$$I\Sigma_1 \vdash \exists t\, \mathsf{T}_{\mathrm{K}}(\overline{p_\delta(x)},0,t) \leftrightarrow \exists w\, \neg \mathsf{Sat}_{\Delta_0}(\bar{d},\bar{x},w).$$

Since $s_\delta(x) = \langle p_\delta(x), 0\rangle$, classical logic yields

$$I\Sigma_1 \vdash \forall t\, U(\overline{s_\delta(x)},t) \leftrightarrow \forall w\, \mathsf{Sat}_{\Delta_0}(\bar{d},\bar{x},w).$$

By Lemma A.2,

$$I\Sigma_1 \vdash \forall w\, \mathsf{Sat}_{\Delta_0}(\bar{d},\bar{x},w) \leftrightarrow \forall w\, \delta(\bar{x},w).$$

Combining the two equivalences gives

$$I\Sigma_1 \vdash \forall w\, \delta(\bar{x},w) \leftrightarrow \forall t\, U(\overline{s_\delta(x)},t).$$

For the final clause, let φ be a Π_1-sentence and write

$$\varphi \equiv \forall w\, \delta(w)$$

with δ bounded. Apply the previous argument to the bounded formula

$$\delta^*(v,w) :\Longleftrightarrow \delta(w),$$

taking $x = 0$. □

Definition 5.2 (Universal Π_1-Code Map and Universal Witness Class) Fix the primitive recursive predicate $U(e,t)$ from Theorem 5.1. Let $\mathsf{Code}_{\Pi_1}(e)$ be the Gödel code of the arithmetic sentence

$$\forall u\, U(\bar{e}, u).$$

Then $e \mapsto \mathsf{Code}_{\Pi_1}(e)$ is primitive recursive.

Define the universal Π^0_1 witness class

$$\mathsf{True}^U_{\Pi_1} := \{e \in \mathbb{N} : \forall t\, U(e,t)\}.$$

Theorem 5.3 (Completeness of the Universal Π^0_1 Witness Class) *The set* $\mathsf{True}^U_{\Pi_1}$ *is* Π^0_1-*complete under primitive recursive injective many-one reducibility. In particular,* $\mathsf{True}^U_{\Pi_1}$ *is undecidable.*

Proof Membership in Π^0_1 is immediate:

$$e \in \mathsf{True}^U_{\Pi_1} \iff \forall t\, U(e,t),$$

and U is primitive recursive.

For hardness, let

$$A = \{x : \forall w\, R(x,w)\}$$

be any Π^0_1 set with R primitive recursive. For each $x \in \mathbb{N}$, define a partial computable machine N_x that on input 0 searches for some w such that $\neg R(x,w)$, halting as soon as one is found and diverging otherwise.

By the *s-m-n* theorem and padding, there exists a primitive recursive injective map

$$p : \mathbb{N} \to \mathbb{N}$$

such that $\varphi_{p(x)}$ computes N_x. Define

$$f(x) := \langle p(x), 0\rangle.$$

Then f is primitive recursive and injective, and

$$x \in A \iff \forall w\, R(x,w) \iff \forall t\, \neg \mathsf{T}_{\mathrm{K}}(p(x),0,t) \iff \forall t\, U(f(x),t) \iff f(x) \in \mathsf{True}^U_{\Pi_1}.$$

Thus $A \leq_m \mathsf{True}^U_{\Pi_1}$ via a primitive recursive injective reduction. ☐

Remark 5.4 Three immediate consequences of Theorem 5.3 are worth recording. First, $\mathsf{True}^U_{\Pi_1}$ has no computable characteristic function. Second, there is no decidable relation $C(e,c)$ such that

$$e \in \mathsf{True}^U_{\Pi_1} \iff \exists c\, C(e,c).$$

Third, for no decidable $W(e,c)$ is there a partial computable f such that

$$e \in \mathsf{True}^U_{\Pi_1} \iff f(e)\downarrow \,\wedge\, W(e, f(e)).$$

The second and third points also follow from Theorems 3.8 and 3.10.

Definition 5.5 (Provability Predicate) Let $T \supseteq I\Sigma_1$ be a recursively axiomatizable arithmetic theory. Fix a standard Σ_1 provability predicate $\mathrm{Prov}_T(x)$ satisfying the Hilbert–Bernays–Löb derivability conditions in T.

Convention 5.6 (Reflection notation) For a recursively axiomatizable arithmetic theory T, write $\mathrm{RFN}_{\Pi_1}(T)$ for the scheme

$$\mathrm{Prov}_T(\ulcorner\varphi\urcorner) \to \varphi$$

as φ ranges over all Π_1-sentences.

Proposition 5.7 (Fragment Reflection from Terminality Adequacy) *Let $T \supseteq I\Sigma_1$ be recursively axiomatizable, let $\mathcal{F}$ be a set of arithmetic sentences, let f assign to each code $\ulcorner\sigma\urcorner$ with $\sigma \in \mathcal{F}$ a natural number $f(\ulcorner\sigma\urcorner)$, and let $\mathsf{Term}(x)$ be an arithmetic formula. Suppose that for every $\sigma \in \mathcal{F}$,*

$$T \vdash \mathrm{Prov}_T(\ulcorner\sigma\urcorner) \to \mathsf{Term}(\overline{f(\ulcorner\sigma\urcorner)}),$$

and

$$T \vdash \mathsf{Term}(\overline{f(\ulcorner\sigma\urcorner)}) \to \sigma.$$

Then for every $\sigma \in \mathcal{F}$,

$$T \vdash \mathrm{Prov}_T(\ulcorner\sigma\urcorner) \to \sigma.$$

Proof For each $\sigma \in \mathcal{F}$, compose the two displayed implications. □

Definition 5.8 (Π_1-Adequacy Along an Embedding) Let $T \supseteq I\Sigma_1$ be recursively axiomatizable, and let $f : \mathbb{N} \to \mathbb{N}$ be primitive recursive. An arithmetic formula $\mathsf{Term}(x)$ is *T-adequate along f* if for every $e \in \mathbb{N}$,

$$T \vdash \mathrm{Prov}_T(\overline{\mathsf{Code}_{\Pi_1}(e)}) \to \mathsf{Term}(\overline{f(e)}), \tag{5.1}$$

$$T \vdash \mathsf{Term}(\overline{f(e)}) \to \forall u\, U(\bar{e}, u). \tag{5.2}$$

Theorem 5.9 (Reflection Collapse Theorem) *Let $T \supseteq I\Sigma_1$ be recursively axiomatizable, and let Prov_T satisfy the derivability conditions. If $\mathsf{Term}(x)$ is T-adequate along some primitive recursive map f, then T proves the full Π_1-reflection scheme*

$$\mathrm{RFN}_{\Pi_1}(T).$$

Proof Let φ be any Π_1-sentence. By Theorem 5.1, choose $e \in \mathbb{N}$ such that

$$I\Sigma_1 \vdash \varphi \leftrightarrow \forall u\, U(\bar{e}, u).$$

Hence T proves the same equivalence.

From

$$T \vdash \varphi \to \forall u\, U(\bar{e}, u),$$

the derivability conditions yield

$$T \vdash \mathrm{Prov}_T(\ulcorner\varphi\urcorner) \to \mathrm{Prov}_T(\overline{\mathsf{Code}_{\Pi_1}(e)}).$$

By (5.1),

$$T \vdash \mathrm{Prov}_T(\overline{\mathsf{Code}_{\Pi_1}(e)}) \to \mathsf{Term}(\overline{f(e)}).$$

By (5.2),

$$T \vdash \mathsf{Term}(\overline{f(e)}) \to \forall u\, U(\bar{e}, u).$$

And since

$$T \vdash \forall u\, U(\bar{e}, u) \to \varphi,$$

we obtain

$$T \vdash \mathrm{Prov}_T(\ulcorner \varphi \urcorner) \to \varphi.$$

Because φ was arbitrary, T proves $\mathrm{RFN}_{\Pi_1}(T)$. $\square$

Corollary 5.10 (Same-Theory Internal Certification Barrier) *Let $T \supseteq I\Sigma_1$ be consistent and recursively axiomatizable. Then no arithmetic formula $\mathsf{Term}(x)$ can be T-adequate along any primitive recursive map f.*

Proof If such Term existed, then by Theorem 5.9,

$$T \vdash \mathrm{RFN}_{\Pi_1}(T).$$

Instantiating reflection at the false Π_1 sentence $0 = 1$, one gets

$$T \vdash \mathrm{Prov}_T(\ulcorner 0 = 1 \urcorner) \to 0 = 1.$$

Since $T \supseteq \mathsf{Q}$ proves $\neg(0 = 1)$, it follows that

$$T \vdash \neg\mathrm{Prov}_T(\ulcorner 0 = 1 \urcorner),$$

that is, T proves its own consistency. This contradicts Gödel's second incompleteness theorem for consistent recursively axiomatizable theories extending $I\Sigma_1$. $\square$

Definition 5.11 (Totality Sentences) For each $a \in \mathbb{N}$, let

$$\theta_a :\Longleftrightarrow \forall n\, \exists m\, \mathsf{T}_{\mathrm{K}}(a, n, m).$$

Thus $a \in \mathsf{TOT}$ iff $\mathbb{N} \models \theta_a$.

Lemma 5.12 (Uniform Coding of Π_1-Sentences by Totality Sentences) *There exists a primitive recursive function $g : \mathbb{N} \to \mathbb{N}$ such that for every $e \in \mathbb{N}$,*

$$I\Sigma_1 \vdash \forall n \left(U(\bar{e}, n) \leftrightarrow \exists t\, \mathsf{T}_{\mathrm{K}}(\overline{g(e)}, n, t) \right).$$

Consequently,

$$I\Sigma_1 \vdash \forall n\, U(\bar{e}, n) \leftrightarrow \theta_{g(e)}.$$

Proof Apply Lemma A.4 to the fixed primitive recursive predicate

$$R(e, n) \;:\Longleftrightarrow\; U(e, n).$$

Let $g := g_R$. Then for every $e, n \in \mathbb{N}$,

$$I\Sigma_1 \vdash U(\bar{e}, \bar{n}) \leftrightarrow \exists t\, \mathsf{T}_{\mathrm{K}}(\overline{g(e)}, \bar{n}, t).$$

Universally quantifying over n yields the claim. $\square$

6 Truth-Faithful Embeddings, the Tarski Threshold, and Diagonal Collapse

Convention 6.1 (Truth notation) For a code y of an arithmetic sentence, $\mathsf{Truth}(y)$ means metamathematically that the sentence coded by y is true in $\mathbb{N}$.

Definition 6.2 (Primitive Recursive Sentence Fragment) A *primitive recursive sentence fragment* is a primitive recursive set

$$\mathcal{F} \subseteq \mathbb{N}$$

of Gödel codes of arithmetic sentences. Write

$$\mathsf{Truth}(\mathcal{F}) := \{y \in \mathcal{F} : \mathsf{Truth}(y)\}.$$

Definition 6.3 (Witness Class and Semantic Success) A *witness class* is a set $\mathcal{N} \subseteq \mathbb{N}$ of codes together with a semantic success predicate

$$\mathsf{Suc}(e)$$

on $\mathcal{N}$.

Definition 6.4 (Truth-Faithful Witness Embedding) Let $(\mathcal{N}, \mathsf{Suc})$ be a witness class and let $\mathcal{F}$ be a primitive recursive sentence fragment. A primitive recursive injection

$$f : \mathcal{F} \to \mathcal{N}$$

is a *truth-faithful witness embedding* if for every $y \in \mathcal{F}$,

$$\mathbb{N} \models \mathsf{Suc}(f(y)) \iff \mathsf{Truth}(y).$$

Definition 6.5 (Exact Arithmetic Terminality Predicate on the Embedded Image) Let $f : \mathcal{F} \to \mathcal{N}$ be a truth-faithful witness embedding. An arithmetic formula $\mathsf{Term}(x)$ is *exact on the embedded image* $f[\mathcal{F}]$ if for every $y \in \mathcal{F}$,

$$\mathbb{N} \models \mathsf{Term}(f(y)) \leftrightarrow \mathsf{Suc}(f(y)).$$

Theorem 6.6 (Exact Threshold Theorem) *Let $f : \mathcal{F} \to \mathcal{N}$ be a truth-faithful witness embedding. Then the following are equivalent:*

(i) there exists an arithmetic formula $\mathsf{Term}(x)$ exact on $f[\mathcal{F}]$;

(ii) the truth set $\mathsf{Truth}(\mathcal{F})$ is arithmetical.

Proof Assume (i), and let $\mathsf{Term}(x)$ be arithmetic and exact on $f[\mathcal{F}]$. Let $F_f(y, z)$ be the fixed graph formula for f from Convention 2.2. Then

$$y \in \mathsf{Truth}(\mathcal{F}) \iff \mathcal{F}(y) \wedge \exists z \big(F_f(y, z) \wedge \mathsf{Term}(z)\big).$$

Because $\mathcal{F}$, F_f, and Term are arithmetic, the right-hand side is arithmetical. Hence $\mathsf{Truth}(\mathcal{F})$ is arithmetical.

Conversely, assume (ii). Let $\mathsf{Tr}_{\mathcal{F}}(y)$ be an arithmetic formula defining $\mathsf{Truth}(\mathcal{F})$. Let $F_f(y, x)$ be the graph formula for f. Define

$$\mathsf{Term}(x) \;:\iff\; \exists y \Big(\mathcal{F}(y) \wedge F_f(y, x) \wedge \mathsf{Tr}_{\mathcal{F}}(y)\Big).$$

Now let $y_0 \in \mathcal{F}$. Since f is injective,

$$\mathbb{N} \models \mathsf{Term}(f(y_0)) \iff \mathbb{N} \models \mathsf{Tr}_{\mathcal{F}}(y_0) \iff y_0 \in \mathsf{Truth}(\mathcal{F}) \iff \mathbb{N} \models \mathsf{Suc}(f(y_0)).$$

Thus Term is exact on $f[\mathcal{F}]$. □

Remark 6.7 The point of Theorem 6.6 is threshold identification rather than one-sided impossibility. Arithmetic exactness on a truth-faithful image is possible precisely up to the point where the corresponding fragment truth set remains arithmetical.

Corollary 6.8 (Tarski Barrier) *Let $(\mathcal{N}, \mathsf{Suc})$ be a witness class. Suppose there exists a truth-faithful witness embedding*

$$f : \mathsf{Sent}_{\text{arith}} \to \mathcal{N}$$

of the full class of arithmetic sentence codes into $\mathcal{N}$. Then there is no arithmetic formula $\mathsf{Term}(x)$ exact on $f[\mathsf{Sent}_{\text{arith}}]$.

Proof By Tarski's undefinability theorem for arithmetic truth in the fixed arithmetic language, the set of true arithmetic sentence codes is not arithmetical. Apply Theorem 6.6. □

Definition 6.9 (Diagonally Universal Witness Class) A witness class $(\mathcal{N}, \mathsf{Suc})$ is *diagonally universal* if there exists a primitive recursive map

$$d : \{\ulcorner \varphi(x) \urcorner : \varphi \text{ an arithmetic formula with one free variable}\} \to \mathcal{N}$$

such that for every arithmetic formula $\varphi(x)$,

$$\mathbb{N} \models \mathsf{Suc}(d(\ulcorner \varphi \urcorner)) \iff \varphi(\overline{d(\ulcorner \varphi \urcorner)}).$$

Theorem 6.10 (Diagonal Barrier) *If $(\mathcal{N}, \mathsf{Suc})$ is diagonally universal, then there is no arithmetic formula $\mathsf{Term}(x)$ such that*

$$\forall e \in \mathcal{N} \qquad \mathbb{N} \models \mathsf{Term}(e) \leftrightarrow \mathsf{Suc}(e).$$

Proof Assume, toward contradiction, such a $\mathsf{Term}(x)$ exists. Apply diagonal universality to

$$\varphi(x) :\iff \neg\mathsf{Term}(x).$$

Let

$$e := d(\ulcorner \neg\mathsf{Term}(x) \urcorner).$$

Then

$$\mathbb{N} \models \mathsf{Suc}(e) \leftrightarrow \neg\mathsf{Term}(e).$$

But exactness gives

$$\mathbb{N} \models \mathsf{Term}(e) \leftrightarrow \mathsf{Suc}(e).$$

Hence

$$\mathbb{N} \models \mathsf{Term}(e) \leftrightarrow \neg\mathsf{Term}(e),$$

a contradiction. □

Theorem 6.11 (Truth Implies Diagonal Universality) *If a witness class* $(\mathcal{N}, \mathsf{Suc})$ *admits a truth-faithful witness embedding*

$$f : \mathsf{Sent}_{\mathrm{arith}} \to \mathcal{N}$$

of all arithmetic sentence codes, then $(\mathcal{N}, \mathsf{Suc})$ *is diagonally universal.*

Proof Let $F_f(x, z)$ be the fixed graph formula for f from Convention 2.2. By primitive recursive manipulation of Gödel codes, there is a primitive recursive function h such that whenever $u = \ulcorner\varphi(z)\urcorner$ codes an arithmetic formula with one free variable, $h(u)$ is the code of the formula

$$\psi_u(x) \ :\Longleftrightarrow\ \exists z \left(F_f(x, z) \wedge \varphi(z)\right).$$

Let

$$q(u) := \mathsf{Fix}(h(u)).$$

Then q is primitive recursive. If $u = \ulcorner\varphi(z)\urcorner$, let λ_u be the sentence coded by $q(u)$. By Lemma A.5,

$$\mathsf{Q} \vdash \lambda_u \leftrightarrow \exists z \left(F_f(\overline{q(u)}, z) \wedge \varphi(z)\right).$$

Define

$$d(u) := f(q(u)).$$

Then d is primitive recursive by Convention 2.4. Since f is truth-faithful,

$$\mathbb{N} \models \mathsf{Suc}(d(u)) \iff \mathsf{Truth}(q(u)) \iff \lambda_u.$$

By the fixed-point equivalence and the fact that F_f represents the graph of f,

$$\lambda_u \iff \exists z \left(F_f(\overline{q(u)}, z) \wedge \varphi(z)\right) \iff \varphi(\overline{f(q(u))}) \iff \varphi(\overline{d(u)}).$$

Hence

$$\mathbb{N} \models \mathsf{Suc}(d(u)) \iff \varphi(\overline{d(u)}),$$

which is exactly diagonal universality. □

Theorem 6.12 (Internal Biconditional Inconsistency Theorem) *Let* $T \supseteq \mathsf{Q}$ *be an arithmetic theory, let* $f : \mathbb{N} \to \mathbb{N}$ *be any primitive recursive map, and let* $\mathsf{Term}(x)$ *be an arithmetic formula. Suppose that for every arithmetic sentence* σ*,*

$$T \vdash \mathsf{Term}(\overline{f(\ulcorner\sigma\urcorner)}) \leftrightarrow \sigma.$$

Then T *is inconsistent.*

Proof Let $F_f(x, z)$ be the fixed graph formula for f from Convention 2.2. Define

$$\theta(x) \ :\Longleftrightarrow\ \exists z \left(F_f(x, z) \wedge \neg\mathsf{Term}(z)\right).$$

By the diagonal lemma, there exists a sentence λ such that

$$\mathsf{Q} \vdash \lambda \leftrightarrow \theta(\overline{\ulcorner\lambda\urcorner}).$$

Let

$$m := f(\ulcorner\lambda\urcorner).$$

By numeral-correctness of F_f,

$$\mathsf{Q} \vdash \exists z \left(F_f(\overline{\ulcorner\lambda\urcorner}, z) \wedge \neg\mathsf{Term}(z)\right) \leftrightarrow \neg\mathsf{Term}(\bar{m}).$$

Hence

$$T \vdash \lambda \leftrightarrow \neg\mathsf{Term}(\bar{m}).$$

By the assumed biconditional scheme, applied to $\sigma = \lambda$,

$$T \vdash \mathsf{Term}(\bar{m}) \leftrightarrow \lambda.$$

Combining the two displayed equivalences, T proves

$$\lambda \leftrightarrow \neg\lambda,$$

and is therefore inconsistent. □

7 Fixed Propositions and the Bridge Trichotomy

Convention 7.1 (Semantic Exactness in the Fixed-Proposition Sections) From this section onward, unless a background theory is explicitly mentioned, exactness statements for a fixed proposition are truth-equivalences in the standard model $\mathbb{N}$.

Throughout this section, P is a fixed proposition equipped with an exact two-sided decidable witness package:

$$\mathbb{N} \models P \iff \forall n\, \exists c\, R_P(n, c), \qquad \mathbb{N} \models \neg P \iff \exists w\, B_P(w),$$

where $R_P(n, c)$ and $B_P(w)$ are decidable predicates.

Definition 7.2 (Isolated Extensional Bridge Sentence) A sentence B is an *isolated extensional bridge* for the fixed proposition P if

$$\mathbb{N} \models B \to P.$$

Theorem 7.3 (Vacuity of Isolated Extensional Bridges) *Let P be a fixed proposition.*

(i) If $\mathbb{N} \models P$, then every true sentence B is an isolated extensional bridge for P.

(ii) If $\mathbb{N} \models \neg P$, then every false sentence B is an isolated extensional bridge for P.

Consequently, the bare extensional notion of a single bridge sentence for a fixed proposition carries no meaningful complexity content.

Proof Immediate from the truth table for implication. □

Definition 7.4 (Effective Bridge Class) Define

$$\mathsf{Bridge}_P(e) \;:\Longleftrightarrow\; \forall n\, \exists t\, \big(\mathsf{T}_{\mathrm{K}}(e, n, t) \wedge R_P(n, \mathsf{U}_{\mathrm{K}}(t))\big).$$

Let

$$\mathcal{B}_P := \{e \in \mathbb{N} : \mathsf{Bridge}_P(e)\}.$$

Theorem 7.5 (Effective Fixed-Proposition Bridge Theorem) *For a fixed proposition P with exact two-sided decidable witness package:*

(i) $\mathcal{B}_P \in \Pi^0_2$;

(ii) *if* $\mathbb{N} \models \neg P$, *then* $\mathcal{B}_P = \varnothing$;

(iii) *if* $\mathbb{N} \models P$, *then there exists a primitive recursive injective map*

$$f : \mathbb{N} \to \mathbb{N}$$

such that

$$a \in \mathsf{TOT} \iff f(a) \in \mathcal{B}_P.$$

Consequently $\mathcal{B}_P$ *is* Π^0_2*-complete.*

Proof Item (i) is immediate from the definition.

For (ii), assume $\mathbb{N} \models \neg P$. Then the exact witness package yields

$$\mathbb{N} \models \neg\forall n\, \exists c\, R_P(n, c),$$

so there is some n_0 such that

$$\mathbb{N} \models \forall c\, \neg R_P(n_0, c).$$

Hence no e can satisfy the defining condition for $\mathcal{B}_P$, and $\mathcal{B}_P = \varnothing$.

For (iii), assume $\mathbb{N} \models P$. Then

$$\forall n\, \exists c\, R_P(n, c),$$

so the least-search machine

$$n \mapsto \mu c\, R_P(n, c)$$

is a successful seed for the local verifier

$$\mathsf{Cert}_P(n, c) :\iff R_P(n, c).$$

Apply Theorem 4.8 with this verifier and with

$$R(a, n, m) :\iff \mathsf{T}_\mathrm{K}(a, n, m).$$

This yields a primitive recursive injective map f such that

$$\mathsf{Bridge}_P(f(a)) \iff \forall n\, \exists m\, \mathsf{T}_\mathrm{K}(a, n, m),$$

that is,

$$f(a) \in \mathcal{B}_P \iff a \in \mathsf{TOT}.$$

Since TOT is Π^0_2-complete by Proposition A.1, $\mathcal{B}_P$ is Π^0_2-complete. □

Corollary 7.6 (Fixed-Proposition Gödel–Turing Ladder) *Let* P *have an exact two-sided decidable witness package and assume* $\mathbb{N} \models P$. *Then:*

(i) *the effective bridge class* $\mathcal{B}_P$ *is* Π^0_2*-complete;*

(ii) *there exists a primitive recursive injective map* $f : \mathbb{N} \to \mathbb{N}$ *such that*

$$a \in \mathsf{TOT} \iff f(a) \in \mathcal{B}_P;$$

(iii) *if* $T \supseteq I\Sigma_1$ *is a consistent recursively axiomatizable theory, then no arithmetic formula* $\mathsf{Term}(x)$ *can satisfy both*

$$T \vdash \mathrm{Prov}_T(\ulcorner \theta_a \urcorner) \to \mathsf{Term}(\overline{f(a)})$$

and

$$T \vdash \mathsf{Term}(\overline{f(a)}) \to \theta_a$$

for all $a \in \mathbb{N}$.

Proof Items (i) and (ii) are Theorem 7.5.

For (iii), suppose such Term existed. Then by Proposition 5.7,

$$T \vdash \mathrm{Prov}_T(\ulcorner \theta_a \urcorner) \to \theta_a \qquad \text{for all } a \in \mathbb{N}.$$

Let φ be any Π_1 sentence. By Theorem 5.1, choose e such that

$$T \vdash \varphi \leftrightarrow \forall n\, U(\bar{e}, n).$$

By Lemma 5.12,

$$T \vdash \forall n\, U(\bar{e}, n) \leftrightarrow \theta_{g(e)}.$$

Hence

$$T \vdash \varphi \leftrightarrow \theta_{g(e)}.$$

From this equivalence, together with the derivability conditions and the reflection instances already obtained for the θ_a, it follows that

$$T \vdash \mathrm{Prov}_T(\ulcorner \varphi \urcorner) \to \varphi.$$

Because φ was arbitrary, T proves full Π_1 reflection for itself, contradicting Gödel's second incompleteness theorem for consistent recursively axiomatizable extensions of $I\Sigma_1$. ☐

Corollary 7.7 (Direct Π_1-Universal Image Inside the Effective Bridge Class) *Let P have an exact two-sided decidable witness package and assume $\mathbb{N} \models P$. Then there exists a primitive recursive map*

$$h : \mathbb{N} \to \mathbb{N}$$

such that for every $e \in \mathbb{N}$,

$$\mathsf{Bridge}_P(h(e)) \iff \forall n\, U(e, n).$$

Consequently, if $T \supseteq I\Sigma_1$ is a consistent recursively axiomatizable theory, then no arithmetic formula $\mathsf{Term}(x)$ can be T-adequate along h.

Proof Choose a primitive recursive injective map f as in Theorem 7.5 such that

$$\mathsf{Bridge}_P(f(a)) \iff \theta_a.$$

Let g be as in Lemma 5.12, and define

$$h(e) := f(g(e)).$$

Then

$$\mathsf{Bridge}_P(h(e)) \iff \theta_{g(e)} \iff \forall n\, U(e, n),$$

the last equivalence by Lemma 5.12. The final claim follows from Corollary 5.10. ☐

Definition 7.8 (Resolver Class for a Fixed Proposition) Interpret elements of W_e as tagged outputs. Use the tags:

(i) $\langle 0, n, c \rangle$: stage-n positive certificate c;

(ii) $\langle 1, w \rangle$: bad witness w for $\neg P$.

Define

$$\mathsf{Res}_P(e)$$

to mean:

$$\Big[\forall n\, \exists c\, \big(\langle 0, n, c \rangle \in W_e \wedge R_P(n, c)\big)\Big] \;\vee\; \Big[\exists w\, \big(\langle 1, w \rangle \in W_e \wedge B_P(w)\big)\Big].$$

Theorem 7.9 (Unconditional Resolver Seed Theorem) *There exists a computable index* u_P *such that*

$$\mathsf{Res}_P(u_P)$$

holds unconditionally.

Proof Construct u_P to dovetail two searches:

(i) for each stage n, search for some c with $R_P(n,c)$, and enumerate $\langle 0,n,c\rangle$ when found;

(ii) simultaneously search for some w with $B_P(w)$, and enumerate $\langle 1,w\rangle$ when found.

Because R_P and B_P are decidable, this is computable. If P is true, the first branch succeeds at every stage; if P is false, the second branch eventually succeeds. Hence $\mathsf{Res}_P(u_P)$ holds in either case. □

Definition 7.10 (Assertion-Enriched Resolver Enlargement) Define

$$\mathcal{A}_P := \{\langle y,e\rangle : y \in \mathsf{Sent}_{\mathrm{arith}},\ e \in \mathbb{N}\}$$

with semantic success

$$\mathsf{Suc}_P^{\mathrm{ass}}(\langle y,e\rangle) \;:\Longleftrightarrow\; \mathsf{Truth}(y) \wedge \mathsf{Res}_P(e).$$

Theorem 7.11 (Assertion-Enriched Truth-Universal Theorem for a Fixed Proposition) *Let* u_P *be the unconditional resolver seed from Theorem 7.9. Then*

$$f(y) := \langle y, u_P\rangle$$

is a primitive recursive truth-faithful embedding of full arithmetic sentence truth into

$$(\mathcal{A}_P, \mathsf{Suc}_P^{\mathrm{ass}}).$$

Consequently:

(i) no arithmetic formula is exact on the image $f[\mathsf{Sent}_{\mathrm{arith}}]$*;*

(ii) $(\mathcal{A}_P, \mathsf{Suc}_P^{\mathrm{ass}})$ *is diagonally universal;*

(iii) any theory $T \supseteq \mathsf{Q}$ *proving*

$$\mathsf{Term}(\overline{f(\ulcorner\sigma\urcorner)}) \leftrightarrow \sigma \qquad \textit{for all arithmetic sentences } \sigma$$

is inconsistent.

Proof By Theorem 7.9,

$$\mathsf{Res}_P(u_P).$$

Hence

$$\mathsf{Suc}_P^{\mathrm{ass}}(\langle y,u_P\rangle) \iff \mathsf{Truth}(y) \wedge \mathsf{Res}_P(u_P) \iff \mathsf{Truth}(y).$$

So f is a truth-faithful embedding. Item (i) follows from Corollary 6.8; item (ii) follows from Theorem 6.11; and item (iii) follows from Theorem 6.12. □

Theorem 7.12 (Bridge Trichotomy Theorem) *Let* P *have an exact two-sided decidable witness package. Then the bridge notion for* P *splits into three regimes.*

(i) ***Isolated extensional bridge sentences.*** *The bare semantic notion of a single extensional bridge sentence is vacuous by Theorem 7.3.*

(ii) ***Effective object-specific bridge classes.*** *The meaningful stagewise bridge object is the effective bridge class $\mathcal{B}_P$, and*

$$\mathbb{N} \models \neg P \implies \mathcal{B}_P = \varnothing, \qquad \mathbb{N} \models P \implies \mathcal{B}_P \text{ is } \Pi^0_2\text{-complete.}$$

Moreover, if P holds, same-theory adequacy on a direct Π_1-universal image of $\mathcal{B}_P$ yields a Gödel–Turing ladder by Corollaries 7.6 and 7.7.

(iii) ***Assertion-enriched resolver enlargements.*** *The resolver class Res_P has an explicit unconditional seed, and the assertion-enriched enlargement $(\mathcal{A}_P, \mathsf{Suc}_P^{\mathrm{ass}})$ is truth-universal. Hence the Tarski and diagonal barriers apply there by Theorem 7.11.*

Proof Combine Theorem 7.3, Theorem 7.5, Corollary 7.6, Corollary 7.7, Theorem 7.9, and Theorem 7.11. □

8 Calibration and Further Directions

Definition 8.1 (A Trivial Benchmark Proposition) Define

$$P_0 :\Longleftrightarrow \forall n\, \exists c\, (c = 0).$$

Let

$$R_{P_0}(n, c) :\Longleftrightarrow (c = 0), \qquad B_{P_0}(w) :\Longleftrightarrow (0 = 1).$$

Proposition 8.2 (Exact Witness Package for the Benchmark Proposition) *The predicates R_{P_0} and B_{P_0} are decidable. Moreover:*

(i)
$$\mathsf{Q} \vdash P_0 \leftrightarrow \forall n\, \exists c\, R_{P_0}(n, c);$$

(ii)
$$\mathsf{Q} \vdash \neg P_0 \leftrightarrow \exists w\, B_{P_0}(w);$$

(iii)
$$\mathsf{Q} \vdash P_0.$$

Proof Decidability is immediate. Item (i) is by definition. For (iii), Q proves $\exists c(c = 0)$; universal generalization yields P_0. For (ii), use (iii), propositional logic, and the equivalence $\exists w\, (0 = 1) \leftrightarrow 0 = 1$. □

Definition 8.3 (Benchmark Effective Bridge Class) Define

$$\mathcal{B}_{P_0} := \left\{ e : \forall n\, \exists t \big(\mathsf{T}_{\mathrm{K}}(e, n, t) \wedge \mathsf{U}_{\mathrm{K}}(t) = 0\big) \right\}.$$

Proposition 8.4 (Benchmark Calibration) *The benchmark proposition P_0 satisfies all of the following:*

(i) $\mathsf{Q} \vdash P_0$*;*

(ii) its effective bridge class $\mathcal{B}_{P_0}$ is Π^0_2-complete;

(iii) its assertion-enriched resolver enlargement is truth-universal and therefore subject to the Tarski and diagonal barriers.

Proof Item (i) is Proposition 8.2. For (ii), $\mathcal{B}_{P_0}$ is the stagewise-local class associated with

$$\mathsf{Cert}_0(n, c) :\Longleftrightarrow (c = 0).$$

The constant-zero total machine is a successful seed, so by Corollary 4.10 it is Π^0_2-complete. For (iii), apply Theorem 7.11 to P_0. ☐

The benchmark calibration shows that the barrier structure is architectural rather than merely a disguised measure of the difficulty of the underlying proposition. Application papers can therefore treat concrete witness packages as instances of a prior metatheory rather than as the source of that metatheory.

9 Conclusion

The paper isolates a general metatheory of exact witness architectures. The main structural result is the Selection Jump Theorem: in the stagewise-local setting, one successful seed already forces Π^0_2 completeness. Around this theorem lie three further barrier layers: finite stagewise prefixes are uniformly insufficient, undecidable co-c.e. witness architectures resist one-shot positive certification, and sufficiently universal internal certification collapses into Π_1 reflection.

At the semantic level, the Exact Threshold Theorem identifies the precise boundary for arithmetic exactness on truth-faithful images, yielding Tarski and diagonal barriers. For fixed propositions, these results combine into the bridge trichotomy: isolated extensional bridges are vacuous, effective bridge classes are empty or Π^0_2 complete, and assertion-enriched resolver layers are truth-universal. For automated reasoning, the message is that exact certification, proof-producing procedures, and self-verifying recognition schemes are subject to sharp structural limits. The paper isolates those limits abstractly, before one turns to any particular application domain.

A Standard Coding and Completeness Lemmas

Proposition A.1 (Standard Completeness Bases) *Under the reducibility convention of Convention 2.3:*

(i) TOT is Π^0_2-complete;

(ii) K_0 is Σ^0_1-complete.

Proof For (i), membership in Π^0_2 is immediate from

$$e \in \mathsf{TOT} \iff \forall n\, \exists t\, \mathrm{T_K}(e, n, t).$$

Let $A \subseteq \mathbb{N}$ be any Π^0_2 set. Write

$$x \in A \iff \forall n\, \exists m\, R(x, n, m)$$

with R primitive recursive. Define the partial computable machine M_x that, on input n, searches for an m such that $R(x, n, m)$, halting with output 0 when found and diverging otherwise. By the *s-m-n* theorem there is a primitive recursive map f such that $\varphi_{f(x)} = M_x$. Then

$$x \in A \iff f(x) \in \mathsf{TOT}.$$

So $A \leq_m \mathsf{TOT}$.

For (ii), membership in Σ^0_1 is immediate from

$$e \in \mathsf{K}_0 \iff \exists t\, \mathsf{T}_{\mathrm{K}}(e, 0, t).$$

Let $B \subseteq \mathbb{N}$ be any Σ^0_1 set. Write

$$x \in B \iff \exists m\, S(x, m)$$

with S primitive recursive. Define the partial computable machine N_x that, on input 0, searches for an m such that $S(x, m)$, halting with output 0 when found and diverging otherwise. By the *s-m-n* theorem there is a primitive recursive map g such that

$$x \in B \iff \varphi_{g(x)}(0)\downarrow \iff g(x) \in \mathsf{K}_0.$$

So $B \leq_m \mathsf{K}_0$. ☐

Lemma A.2 (Uniform Bounded-Formula Evaluation) *There exists a primitive recursive predicate* $\mathsf{Sat}_{\Delta_0}(d, x, w)$ *such that for every bounded arithmetic formula* $\delta(v, w)$ *whose free variables are among* v, w, *if* $d = \ulcorner \delta(v, w) \urcorner$, *then*

$$I\Sigma_1 \vdash \forall x\, \forall w\, \big(\mathsf{Sat}_{\Delta_0}(\bar{d}, x, w) \leftrightarrow \delta(x, w)\big).$$

Proof This is standard; see [2, 3]. ☐

Lemma A.3 (Search-Template Formalization in $I\Sigma_1$) *Let* $R(x, y)$ *be a fixed primitive recursive predicate. Then there exists a primitive recursive injective function*

$$s_R : \mathbb{N} \to \mathbb{N}$$

such that for every $x \in \mathbb{N}$,

$$I\Sigma_1 \vdash \exists t\, \mathsf{T}_{\mathrm{K}}(\overline{s_R(x)}, 0, t) \leftrightarrow \exists y\, R(\bar{x}, y).$$

Proof Use the *s-m-n* theorem and padding to assign to each x an index of the program that, on input 0, searches $y = 0, 1, 2, \ldots$ until it finds one with $R(x, y)$, and halts immediately when it does so. Because R is primitive recursive, the step-by-step correctness of this search program is representable and provably correct in $I\Sigma_1$. Injectivity is obtained by standard padding. ☐

Lemma A.4 (Conditional-Halting Template Formalization in $I\Sigma_1$) *Let* $R(x, n)$ *be a fixed primitive recursive predicate. Then there exists a primitive recursive function*

$$g_R : \mathbb{N} \to \mathbb{N}$$

such that for every $x, n \in \mathbb{N}$,

$$I\Sigma_1 \vdash R(\bar{x}, \bar{n}) \leftrightarrow \exists t\, \mathsf{T}_{\mathrm{K}}(\overline{g_R(x)}, \bar{n}, t).$$

Proof Again use the *s-m-n* theorem: for each x, let $g_R(x)$ index the program that, on input n, computes the primitive recursive predicate $R(x, n)$, halting immediately if it is true and diverging otherwise. The arithmetized correctness of this fixed program template is provable in $I\Sigma_1$. □

Lemma A.5 (Primitive Recursive Fixed-Point Operator) *There exists a primitive recursive function* Fix *such that whenever* u *codes an arithmetic formula* $\psi(x)$ *with one free variable, the sentence* λ_u *coded by* $\mathsf{Fix}(u)$ *satisfies*

$$\mathsf{Q} \vdash \lambda_u \leftrightarrow \psi(\overline{\mathsf{Fix}(u)}).$$

Proof This is the effective diagonal lemma in uniform form; see [1, 2]. □

Acknowledgements. Not applicable.

Declarations. Funding No external funding was received for this work.

Competing interests The authors declare that they have no competing interests.

Ethics approval and consent to participate Not applicable.

Consent for publication Not applicable.

Data availability No datasets were generated or analysed during the current study.

Materials availability Not applicable.

Code availability Not applicable.

Author contribution Both authors contributed to the conceptual development, theorem formulation, proof design, drafting, and revision of the manuscript. Both authors read and approved the final manuscript.

Use of AI-assisted technologies in writing The authors used generative AI tools for drafting and editorial assistance. All mathematical claims, proofs, and final wording were reviewed and approved by the authors.

References

[1] G. Boolos, J. Burgess, and R. Jeffrey, *Computability and Logic*, 5th edn. (Cambridge University Press, Cambridge, 2007)

[2] P. Hájek and P. Pudlák, *Metamathematics of First-Order Arithmetic* (Springer, Berlin, 1993)

[3] S. C. Kleene, *Introduction to Metamathematics* (North-Holland, Amsterdam, 1952)

[4] H. Rogers, Jr., *Theory of Recursive Functions and Effective Computability* (MIT Press, Cambridge, MA, 1987)

[5] C. Smoryński, *Self-Reference and Modal Logic* (Springer, New York, 1985)

[6] R. I. Soare, *Turing Computability: Theory and Applications* (Springer, Heidelberg, 2016)

[7] A. Tarski, The concept of truth in formalized languages, in: *Logic, Semantics, Metamathematics*, 2nd edn. (Hackett, Indianapolis, 1983)