Article

Not peer-reviewed version

Categorical Analyses of Gödel and Tarski Twin-Theorems

Yingrui Yang * and Hongbin Wang

Posted Date: 31 December 2024

doi: 10.20944/preprints202412.2496.v1

Keywords: category; Gödel; Tarski; first-order theory; syntax; semantics; structure; high-order category; expressibility; definability doi:10.20944/preprints202412.2496.v1

Article

Categorical Analyses of Gödel and Tarski Twin-Theorems

Yingrui Yang [1] and Hongbin Wang [2]

[1] Department of Cognitive Science, Rensselaer Polytechnic Institute
[2] Texas A&M University
* Correspondence: yangyri@rpi.edu

Abstract: This paper provides a set of category-theoretic analyses of Gödel's incompleteness theorems and Tarski's indefinability theorem (see Appendix). We view the first-order theory as a mathematical language and introduce the notion of "language charge" as a monad within a category. For each analysis, we introduce a pair of adjunct categories: a syntactic category and a semantic category. We show that the Gödel numbering can be modeled as a pair of adjoint functors between these categories—a right functor from syntax to semantics and a left functor in the reverse direction. We prove that the Gödel numbering functor serves as a limit in a functor category. Additional analyses focus on the expressibility and definability in the twin theorems. Each of these is linked to natural transformations. In addition, we establish a formal account of "spontaneous naturality breaking" in the context of Gödel's independent statements and Tarski's indefinability. Finally, it touches higher order categories. By composing a syntactic category and a semantic category, we constructed a 2-Category with two layers of structures. Note that 2-category is one of the current research interests in category theory. Further, by decomposing and recomposing the syntactic category, we constructed a 3-Category.

Keywords: Category; Gödel; Tarski; first-order theory; syntax; semantics; structure; high-order category; expressibility; definability

---

1. Mathematical Language as a Monad Structure

In this paper, we presume the first-order theory [4] without delving into its detailed expansion in metamathematics. We conceptualize the first-order theory as a mathematical language and formalize it using a monad structure in category theory, referred to as the Gödel charge. This approach not only simplifies and clarifies the rich structures inherent in Gödel [4] and Tarski's work [5] but also aligns more effectively with category-theoretic characterizations without compromising generality.

The syntactic and semantic aspects of a first-order theory can be captured using two categories: the syntactic category (Γ) and the semantic category ($\mathbb{N}$) [2]. The relationship between these categories is mediated by the Gödel charge, which establishes a correspondence between the formal (syntactic) world and the arithmetic (semantic) world.

1.1. Syntactic Category (Γ)

The syntactic category Γ corresponds to the abstract world of symbols, formulas, and proofs. This category encodes the structure of logical expressions and derivations in a formal system.

Definition 1 (Objects in Γ). The objects of Γ are sorts, formulas, and proofs. Sorts are the basic types from which more complex expressions are built. These types are typically the sorts from the signature Σ. Formulas are logical expressions involving sorts, function symbols, and relation symbols. Formulas are constructed using the logical connectives ($\wedge, \vee, \neg$) and quantifiers ($\forall, \exists$). Proofs

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 December 2024 doi:10.20944/preprints202412.2496.v1

are sequences of formulas, following rules of inference, that establish the logical derivability of one formula from a collection of premises.

Definition 2 (Morphisms in Γ). Morphisms in Γ correspond to formation rules and inference rules. Formation rules describe how to build formulas from terms, function symbols, and relation symbols. Inference rules capture logical entailment. Identity morphisms correspond to tautological proof steps, where a formula entails itself (P ⇒ P). Composition of morphisms reflects the transitivity of logical entailment (if φ ⇒ ψ and ψ ⇒ χ, then φ ⇒ χ).

The syntactic category Γ can be endowed with a monadic structure. This arises from the observation that Γ is a free Cartesian category generated by the signature Σ. The free Cartesian construction naturally defines a monad on the underlying category of sorts. The monad T on the category of sorts is given by the free functor that takes a set of sorts S and produces the set of all finite products, function symbols, and formulas that can be generated from those sorts. The unit of this monad injects a sort into the space of terms (viewing a sort as a trivial term), and the multiplication flattens nested products and formulas. This monadic structure reflects the compositional nature of logical derivations, allowing terms and proofs to be built recursively.

1.2. Semantic Category (ℕ)

The semantic category ℕ captures the concrete meaning of the syntactic structure. While the syntactic world is abstract and symbolic, the semantic world is concrete and typically based on arithmetic interpretations or arithmetic-theoretic models.

Definition 3 (Objects in ℕ). The objects in ℕ are Gödel numbers and arithmetic structures. Gödel numbers encode syntactic objects (like formulas, terms, and proofs) as natural numbers. Arithmetic structures are sets (like ℕ or ℤ) that interpret the sorts from Σ as concrete domains of computation. For example, if A is a sort in Σ, its interpretation in ℕ could be the set ℕ (natural numbers).

Definition 4 (Morphisms in ℕ). Morphisms in ℕ are defined using arithmetic operations and logical rules. These morphisms include functions on Gödel numbers, such as primitive recursive functions. Morphisms also reflect the way logical entailments are mapped into arithmetic statements. Identity morphisms in ℕ are functions that map a Gödel number to itself. Composition of morphisms in ℕ follows the composition of arithmetic functions.

The semantic category ℕ can also be endowed with a monadic structure. The monadic structure on ℕ can be seen as the list monad, where each syntactic object (like a proof) is assigned a Gödel number, and recursive computations on these numbers follow the structure of primitive recursive functions. The monad's unit injects an arithmetic object (like a number) into the computational context, and the multiplication of the monad corresponds to concatenating computational steps.

2. Gödel Numbering as a Functor

Gödel numbering is one of the key techniques used in Gödel's incompleteness theorem [4] and Tarski's indefinability theorem [5]. The necessary backgrounds of Gödel's theorem and Tarski's theorem are given in the appendix. Below we explain the Gödel numbering method. Mathematical language always deals with symbols, formulas, and derivations. For a mathematical framework, even though its base domains (such as real or complex fields) are uncountable infinities (i.e., the continuum), the number of symbols used to denote variables, functions, operators, etc., is infinite but countably many. Thus, we can have an effective procedure to mechanically assign a unique odd number to each and every symbol in order, called *Gödel number*. For a given symbol e, its Gödel number is written as $g(e)$, which can be seen as a function or an odd number. A formula is a finite string of symbols, written as

$$L = e_1 e_2 \ldots \ldots e_n \qquad (1.1)$$

The Gödel number of a formula can be calculated by

$$g(L) = q_1^{g(e_1)} q_2^{g(e_2)} \ldots q_n^{g(e_n)} \qquad (1.2)$$

where q_i is the first i prime numbers in its natural order, and $g(e_i)$ is the Gödel number of the ith symbol in the formula L. A derivation is a finite sequence of formulas, written as

$$Der(L) \equiv < u_1, u_2, \ldots, u_m > \qquad (1.3)$$

The Gödel number of a derivation can be calculated by

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 December 2024 doi:10.20944/preprints202412.2496.v1

$$g(L) = q_1^{g(u_1)} q_2^{g(u_2)} \ldots q_m^{g(u_m)} \qquad (1.4)$$

where $g(u_i)$ is the Gödel number of the *i*th formula in the derivation sequence. The Gödel number of any given formula or derivation is always an even number, which is also a composite number.

The above method is called *Gödel numbering* [4]. The beauty and power of Gödel numbering is that, based on the so-called first theorem of arithmetic (i.e., Pair forming LCM), from a given Gödel number we can uniquely recapture the original derivation, the original formula, or the original symbol used in the context.

Note that logic has nothing to do with the content. So that the *first-order characterization* in terms of category theory only requires three conditions. First, the syntactic components can be represented by a Gödel number. Second, any given derivation of a particular kind and its Gödel number can be used interchangeably. Third, it allows to introduce new predicates or function terms.

Essentially, Gödel numbering encodes symbols, formulas, and proofs into natural numbers. This encoding can be viewed as a functor between the syntactic category Γ and the semantic category ℕ. The functor maps each object in Γ (such as a symbol, formula, or derivation) to its corresponding Gödel number in ℕ. For morphisms in Γ, such as formation and inference rules, the functor assigns arithmetic operations that mirror these logical transformations in ℕ. Consequently, the Gödel numbering functor preserves the logical structure of derivations, enabling a categorical correspondence between syntactic operations and their semantic counterparts.

In addition to this, it is possible to define an adjoint functor from the semantic category ℕ back to the syntactic category Γ. This adjoint functor essentially reconstructs the syntactic representation corresponding to a given Gödel number in ℕ. For each object (a Gödel number) in ℕ, the adjoint functor identifies the corresponding syntactic construct in Γ, such as a symbol, formula, or derivation. Morphisms in ℕ, which are arithmetic operations on Gödel numbers, are mapped to syntactic morphisms like logical inference steps. This adjoint relationship establishes a bidirectional correspondence between the syntactic and semantic realms, reinforcing the duality that underpins Gödel's incompleteness theorem.

This adjunction can be captured via the concept of Gödel charge, which connects the syntactic world (Γ) to the semantic world (ℕ) via monadic structures. Specifically, the Gödel charge is a monadic natural transformation from the monad on Γ to the monad on ℕ. This transformation maps objects in the syntactic monad (like terms and proofs) to objects in the semantic monad (Gödel numbers). It reflects how the composition of terms in Γ maps to the composition of arithmetic operations on Gödel numbers in ℕ.

3. Gödel Numbering as A Limit of Functors

We have now defined the Gödel functor G: Γ → ℕ and its adjoint G*: ℕ → Γ. In fact, all possible functors from the syntactic category Γ to the semantic category ℕ form a new category, denoted as Fun(Γ, ℕ). We now demonstrate that the Gödel functor G serves as a limit of this category [1], meaning that it is the most "concise and informationally complete" functor among all possible functors in Fun(Γ, ℕ).

A functor G: Γ → ℕ is said to be a limit of a category of functors Fun(Γ, ℕ) if for every functor F: Γ → ℕ, there exists a unique natural transformation η: F → G. This property implies that every functor in Fun(Γ, ℕ) factors through G in a unique way, making G a universal recipient for natural transformations from any other functor F to ℕ.

To understand why the Gödel functor serves as a limit, consider how it interacts with the syntactic and semantic categories. The objects of Γ include symbols, strings of symbols (formulas), and derivations. The Gödel functor G maps each of these objects to their Gödel numbers in ℕ. For morphisms (which correspond to formation rules and inference rules) in Γ, the Gödel functor applies arithmetic transformations to the corresponding Gödel numbers.

Suppose F: Γ → ℕ is an arbitrary functor. For every object X in Γ, F(X) is an object in ℕ. To define a natural transformation η: F → G, we assign to each object X ∈ Γ a morphism η_X: F(X) → G(X) in ℕ. By the definition of Gödel numbering, each element in ℕ corresponds to a unique syntactic construct in Γ. Thus, for each X ∈ Γ, there is a canonical mapping from F(X) to G(X), since G explicitly tracks the syntactic structure of X via its Gödel number.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 December 2024 doi:10.20944/preprints202412.2496.v1

The uniqueness of η follows from the fact that the Gödel numbering scheme is injective and reflects the structure of Γ in a complete and lossless manner. In essence, every syntactic structure in Γ has a unique image in ℕ via the Gödel numbering. Any other functor F mapping from Γ to ℕ can be related to G by a natural transformation, but since G retains all the structural information, the transformation is unique. In formal terms, for every object X ∈ Γ, we have a commutative diagram of the form:

$$\begin{array}{ccc} F(X) & \xrightarrow{F(f)} & F(Y) \\ \downarrow \eta_X & & \downarrow \eta_Y \\ G(X) & \xrightarrow{G(f)} & G(Y) \end{array}$$

for every morphism f: X → Y in Γ. This commutative square ensures that the transformation from F to G respects the morphisms in Γ and is, therefore, a natural transformation.

This universal property [2] highlights the fundamental role of Gödel numbering as a canonical method for encoding syntactic objects as semantic objects. In the categorical perspective, G captures the essence of the Gödel numbering process, and its status as a limit reflects its role as the most "informationally complete" mapping from syntax to semantics.

4. The Expressibility Functor

Definition 5 (Expressibility). In the first-order theory, if the relation $R(a_1, \ldots, a_n)$ holds in $\boldsymbol{N}$, then $P(\boldsymbol{a_1}, \ldots, \boldsymbol{a_n})$ is provable in $\boldsymbol{\mathcal{N}}$. If the relation $R(a_1, \ldots, a_n)$ does not hold in $\boldsymbol{N}$, then $\neg P(\boldsymbol{a_1}, \ldots, \boldsymbol{a_n})$ is provable in $\boldsymbol{\mathcal{N}}$.

We can write this definition in a fully symbolic format:

(i) $\vDash_{\mathrm{N}} R(a_1, \ldots, a_n) \longrightarrow \vdash_{\mathcal{N}} \mathrm{P}(\boldsymbol{a_1}, \ldots \boldsymbol{a_n})$.

(ii) $\nvDash$ N $R(a_1, \ldots, a_n) \longrightarrow \vdash_{\mathcal{N}} \neg\mathrm{P}(\boldsymbol{a_1}, \ldots \boldsymbol{a_n})$.

Now we can define two categories in an abstract but simple way:

Definition 6. The Category $\mathcal{A}$ contains three objects and two morphism arrows below:

Object 1. $\mathrm{M_A}$ (the syntactic monad stands for the language of the first-order theory);

Object 2. $\vdash_{\mathcal{N}} \mathrm{P}(\boldsymbol{a_1}, \ldots \boldsymbol{a_n})$;

Object 3. $\vdash_{\mathcal{N}} \neg\mathrm{P}(\boldsymbol{a_1}, \ldots \boldsymbol{a_n})$.

Arrow 1. The reducing arrow from an object to the monad.

Arrow 2. The contradiction arrow between Object 2 and Object 3.

Definition 7. Category $\mathcal{B}$ contains three objects and two morphism arrows below:

Object 1. $\mathrm{M_B}$ (the semantic monad stands for the language of the first-order theory);

Object 2. $\vDash_{\mathrm{N}} \mathrm{R}(a_1, \ldots, \mathrm{a_n})$;

Object 3. $\nvDash$ $\mathrm{N}\mathrm{R}(a_1, \ldots, a_n)$;

Arrow 1. The reducing arrow from an object to the monad.

Arrow 2. The contradiction arrow between Object 2 and Object 3.

Now we can define the Experessibility functor from Category $\mathcal{B}$ to Category $\mathcal{A}$:

(i) $\vDash_{\mathrm{N}} R(a_1, \ldots, a_n) \longrightarrow \vdash_{\mathcal{N}} \mathrm{P}(\boldsymbol{a_1}, \ldots \boldsymbol{a_n})$;

(ii) $\nvDash_{\mathrm{N}} R(a_1, \ldots, a_n) \longrightarrow \vdash_{\mathcal{N}} \neg\mathrm{P}(\boldsymbol{a_1}, \ldots \boldsymbol{a_n})$;

(iii) $M_B \longrightarrow M_A$.

This functor fully captures the information in the definition of expressibility. We denote this functor as F. Obviously, this is a logical factor. Interestingly, from the perspectives of category theory, we may also construct an anti-logic functor from $\mathcal{B}$ to $\mathcal{A}$ as follows.

Definition. An anti-logic functor, denoted as D, is defined as:

(i) $\vDash_{\mathrm{N}} R(a_1, \ldots, a_n) \longrightarrow \vdash_{\mathcal{N}} \neg\mathrm{P}(\boldsymbol{a_1}, \ldots \boldsymbol{a_n})$;

(ii) $\nvDash_{\mathrm{N}} R(a_1, \ldots, a_n) \longrightarrow \vdash_{\mathcal{N}} \mathrm{P}(\boldsymbol{a_1}, \ldots \boldsymbol{a_n})$;

(iii) $M_{\mathrm{B}} \longrightarrow M_{\mathrm{A}}$.

Actually, D is well-defined in terms of category theory. Obviously, there exists a natural transformation from D to F. Here the meaning of the naturality tells us not only the difference between being logical and anti-logical, but also the relation of them. Being logical or being anti logical are both based on what we mean by logic. However, this question goes beyond the scope of category theory. The natural transformation (from *D* to *F*) also indicates that category theory focuses on the abstract structures. Any contents are only represented by objects and arrows.

5. The Definability Functor and Functor Breaking

Consider a mother statement $P(x)$ and assume its Gödel number $g(P(x)) = m$. We may construct a self-reflective daughter statement $S = P(^m/_x)$, assume $g(S(m)) = n$. Given this, we can introduce a semantic relation $d(m, n)$. According, we can syntactically have $D(\boldsymbol{m}, \boldsymbol{n})$. Tarski introduced the notion of definability, which says that $d(m, n)$ defines $D(\boldsymbol{m}, \boldsymbol{n})$.

Now we can introduce the definability functor, denoted as *D*. Let $\mathcal{A}$ be the syntactic category, which contains a monad and a family of objects in forms of $D(\boldsymbol{m}, \boldsymbol{n})$. Let B be the semantic category, which contain a monad and a family of objects in form of $d(m, n)$. Then we can define a definability functor from $\mathcal{B}$ to $\mathcal{A}$ with map below:

(*i*) the monad in $\mathcal{B}$ ⟶ the monad in $\mathcal{A}$;

(*ii*) each $d(m, n)$ in $\mathcal{B}$ ⟶ the corresponding $D(\boldsymbol{m}, \boldsymbol{n})$ in $\mathcal{A}$.

Obviously, this is a well-defined one-to-one mapping. However, a problem will occur shortly.

Tarski constructed a statement with a free variable *x* below:

$P(x) = \forall y(D(x, y) \to \neg T(y))$ (5.1)

Assume its Gödel number $g(P(x)) = m$. We can construct a self-reflective statement,

$L = P(^m/_x)$, by substituting *m* for *x*. Assume its Gödel number $g(L) = n$. Thus, by the definition earlier, we can have $d(m, n)$, which defines $D(\boldsymbol{m}, \boldsymbol{n})$. In this sense, we say $d(m, n)$ is a model of $D(\boldsymbol{m}, \boldsymbol{n})$. By (5.1), we can logically infer ∀y¬T(y). If the truth predicate T is definable, it supposes to have a model

$X = \{\ L_g \mid X \vDash L_T$, meaning L_T is presupposed to be true under $X\}$.

Tarski' indefinability theorem (see Appendix) show that this mode *X* is null. Thus, predicate *T* is indefinable. Notice that *T*(*y*) is part of original mother formula *P*(*x*). Hence, the indefinability of *T* is carried by the Gödel number *m* of *P*(*x*), and this information is transferred in to the daughter formulas L and its Gödel number *n*, and continue to be transferred into $d(m, n)$ and $D(\boldsymbol{m}, \boldsymbol{n})$. In other words, the definability functor *D* contains the information of indefinability of *T*. We refer this phenomenon as *spontaneous functor breaking*.

6. Gödel Proof and Independent Statement

Gödel theorem (1931) has a special but interesting structure, which can be characterized in category theory. Gödel theorem involves two categories, the syntactic category $\mathcal{A}$ and the semantic category $\mathcal{B}$, and the Gödel theorem itself serves as a functor between the two categories and it also causes the spontaneous functor breaking.

The category $\mathcal{A}$ contains the following objects: a monad and a family of triplets, denoted as $<O_1, O_2, O_3>$, which is defined as follows.

O_1 stands for a statement with one free variable, write *P*(*x*), called the mother sentence.

O_2 stands for a self-reflective statement $S = P(^i/_x)$, where $i = g(P(x))$.

O_3 stands for the proof of *S*, denoted as *Bew*(*S*).

$\mathcal{A}$ contains only two families of morphism arrows. One is the identity arrow of each triplet, and the other is from any triplet to the monad.

The category $\mathcal{B}$ contains the following objects: a monad and a family of ordered triplets, denoted as $< g_1, g_2, g_3 >$, which are defined as below: $g_1 = g(P(x))$; $g_2 = g(S)$; $g_3 = g(Bew(S))$. $\mathcal{B}$ contains two morphism arrows. One is from a mother statement *P*(*x*) to its mother statement *S*, and the other is from *S* to *Bew*(*S*). $\mathcal{B}$ contains only two families of morphism arrows. One is the identity arrow of each triplet, and the other is from any triplet to the monad.

It is obvious that there is a one-to-one bidirectional mapping from A to B and *vice versa*. We name it as the Gödel proof functor. However, there is a problem up to this point. Gödel constructed a statement as below:

$P(x) = \forall y \neg G(x, y)$ (6.1)

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 December 2024 doi:10.20944/preprints202412.2496.v1

Gödel found (1931) that for this particular mother formula, $P(x)$, in its corresponding triplet $<O_1, O_2, O_3>$, O_3 is null; i.e., $Bew(S)$ does not exist! This means that it is no model triplet $< g_1, g_2, g_3 >$ in $\mathcal{B}$. We refer this phenomenon as *spontaneous proof breaking*. Gödel also found the same situation for $\neg S$. Thus, S is called an independent statement.

7. Higher Order Categories

Consider the syntactic category Γ and the semantic category ℕ introduced in Section 1. We can define a new general category Π, which contains two objects Γ and ℕ. We know that the objects in Γ are one-one-corresponding to their Gödel numbers in ℕ. Let C be the Cartesian product of Γ and ℕ [3], we have

$C = \Gamma \times \mathbb{N} = \{(x, g(x))\}$

where $x \in \Gamma$ and $g(x) \in \mathbb{N}$. Thus, the newly introduced general category Π contains two objects and one morphism. This is the first structural layer.

From Section 1, Γ and ℕ are well-defined categories, each with its own structures. This is the second structural layer. In other words, the new category Π has two layers of structures. The category of this kind is called 2-Category [1]. Notice that for the 2-category Π, the arrows in the first layer structure and the second layer are all morphisms but not functors.

Similar treatment can be applied to the category Γ alone. As defined earlier, Γ contains three families of objects, namely, symbols, well-formed-formulas, and proofs. It also contains two arrows, one is for formation rules and the other for inference rules. Now we decompose Γ into two categories. Γ^1 contains only two families of objects, symbols and formulas, and one arrow stands for the formation rules. Γ^2 contains two families of objects, formulas and proofs, and one arrow stands for inference rules. Then, by composing Γ^1 and Γ^2, it yields a new category Γ', which add a third layer of structures to the grandmother category Π, write Π'. Up to this point, we may refer Π' as a 3-category.

8. Conclusions

In this paper, we have provided a unified categorical treatment of Gödel's incompleteness theorem and Tarski's indefinability theorem. By treating first-order logic as categories, we introduced the concept of Gödel charge as a monadic structure and demonstrated how syntactic and semantic categories interact via Gödel numbering, expressibility, and definability. The main contributions are summarized as follows:

Gödel charge: The syntactic category has a free Cartesian monad for constructing terms and formulas, while the semantic category has a list monad on Gödel numbers. The Gödel charge acts as a monadic natural transformation linking the two.

Gödel Numbering as a Limit Functor: We established that Gödel numbering is a limit functor between the syntactic category Γ and the semantic category ℕ. This functor serves as the most complete mapping, preserving all syntactic information.

Expressibility and Definability Functors: We formalized the concepts of expressibility and definability as categorical functors. While the expressibility functor retains natural transformations for provability, the definability functor experiences spontaneous functor breaking, reflecting Tarski's theorem on the indefinability of truth as a predicate.

Spontaneous Functor Breaking: We formalized how the definability and proof functors experience "spontaneous breaking" when dealing with self-referential statements. This concept provides a categorical insight into the limitations of formal systems in representing truth and provability.

Higher Order Category: By composing a syntactic category and a semantic category, we constructed a 2-category with two layers of structures. Note that 2-category is one of the current research interests in category theory. By further decomposing and recomposing a syntactic category, we constructed a 3-category with three layers of structures.

By unifying Gödel's and Tarski's theorems through the lens of category theory, we offer a new perspective on the duality between syntax and semantics. The framework of natural transformations, adjoint functors, and functor limits serves as a robust mathematical foundation for exploring the intrinsic limitations of formal logical systems.

Acknowledgments: The authors thank Zhangju Liu for helpful discussions and insightful comments.

Appendix. Twin Theorems of Gödel and Tarski [6]

We assume readers are familiar with the first order logic (**PL**), the first order theory ($\boldsymbol{\mathcal{N}}$), and the arithmetic theory (N). Intuitive natural numbers used in N are given by n, and the corresponding enumerers used in $\boldsymbol{\mathcal{N}}$ are denoted by bold $\boldsymbol{n}$. Enumerers are constructed by starting from the empty set $\emptyset$ and the so-called successor function, such that $\emptyset = \mathbf{0}$, $\{\emptyset\} = \mathbf{1}$, $\{\emptyset, \{\emptyset\}\} = \mathbf{2}$, and so forth. In the following, we introduce Gödel's theorem first, and Tarski's theorem second.

Definition of Expressibility: If $R(a_1, \dots, a_n)$ holds in $\boldsymbol{N}$, then $P(\boldsymbol{a_1}, \dots, \boldsymbol{a_n})$ is provable in $\boldsymbol{\mathcal{N}}$. If $R(a_1, \dots, a_n)$ does not hold in $\boldsymbol{N}$, then $\neg P(\boldsymbol{a_1}, \dots, \boldsymbol{a_n})$ is provable in $\boldsymbol{\mathcal{N}}$.

Definition of Consistency: For any given formula "L" in $\boldsymbol{\mathcal{N}}$, either L is provable, or else $\neg L$ is provable, but not both.

For a given formula L, denote its proof by $Bew(L)$. Assume $g(L) = i$ and $g(Bew(L)) = j$, where i and j are Gödel numbers. We introduce a relation $\mathcal{G}(i, j)$ in $\boldsymbol{N}$, and define a function term $G(\boldsymbol{i}, \boldsymbol{j})$ in $\boldsymbol{\mathcal{N}}$. Gödel constructed a formula,

$P(x) = \forall y \neg G(x, y)$,

in which x is a free variable. Let $g[P(x)] = i$, by substituting x with i, we can use,

$S = P(\boldsymbol{i}) = \forall y \neg G(\boldsymbol{i}, y)$

This is a so-called self-reflection sentence.

Gödel First Theorem Neither S nor $\neg S$ is provable in $\boldsymbol{\mathcal{N}}$.

We now briefly sketch a proof. First, we prove that S is not provable. Assume for contradiction that S is provable, then it must have a proof, write $Bew(S)$, let $g(Bew(L)) = j$ and $g(S) = i$, so that $\mathcal{G}(i, j)$ in $\boldsymbol{N}$. By the expressibility, $G(\boldsymbol{i}, \boldsymbol{j})$ must be provable in $\boldsymbol{\mathcal{N}}$; but S said that for any $\boldsymbol{j}$, $\neg G(\boldsymbol{i}, \boldsymbol{j})$. This contradiction shows that the assumption is impossible. Hence, S is not provable in $\boldsymbol{\mathcal{N}}$.

Second, we prove that $\neg S$ is unprovable in $\boldsymbol{\mathcal{N}}$. Assume for contradiction that $\neg S$ is provable. Then by consistency, S is unprovable, so that for any j, $g(Bew(S)) \neq j$. Hence, for any j, $\mathcal{G}(i, j)$ does not hold in $\boldsymbol{N}$; by expressibility, $\neg G(\boldsymbol{i}, \boldsymbol{j})$, for any j. As such, by $\omega\_consistency$, we have $\forall y \neg G(\boldsymbol{i}, y)$, which means S is provable in $\boldsymbol{\mathcal{N}}$. This contradicts to the assumption that S is unprovable. Thus $\neg S$ is unprovable in $\boldsymbol{\mathcal{N}}$.

The above result shows that the consistency of $\boldsymbol{\mathcal{N}}$ is independent of $\boldsymbol{\mathcal{N}}$. Now let us speculate about what S expresses. S is a self-reflection sentence, it says that S is unprovable, and we have just proved it above; thus, *S is true, but not provable in $\boldsymbol{\mathcal{N}}$*, which by definition means that $\boldsymbol{\mathcal{N}}$ is incomplete. This is the well-known Gödel Incompleteness Theorem. We now turn to Tarski's indefinability theorem.

Definition of Definability: Let $g(P(x)) = m$, and $g(P(m)) = n$, we can hold a binary relation $d(m, n)$ in $\boldsymbol{N}$. Accordingly, we say $D(\boldsymbol{m}, \boldsymbol{n})$ is definable in $\boldsymbol{\mathcal{N}}$, meaning $D(\boldsymbol{m}, \boldsymbol{n})$ has a model, which is not null.

Tarski introduced a new predicate of being true, denoted by $\boldsymbol{T}$, and he constructed a sentence below:

$A(x) = \forall y[D(x, y) \to \neg T(y)]$

Let $g[A(x)] = m$, substituting x by m, we have

$B(\boldsymbol{m}) = \forall y[D(\boldsymbol{m}, y) \to \neg T(y)]$

Let $g[B(m)] = n$, we have $d(m, n)$ which holds in $\boldsymbol{N}$; hence, $D(x, y)$ is definable in $\boldsymbol{\mathcal{N}}$. Then, by standard logic, we can infer $\neg T(y)$. Now we show that T is not definable, meaning its model is null. Denote $B(\boldsymbol{m})$ by L. If L is pre-assumed as true, denote it by Lt, and write $Lg = g(L)$. As such, we may assume for contradiction that T had a model X;

$X = \{ Lg \mid X \vDash L;$ i.e., Lt is presupposed to be true under $X\}$

Since $Lg = g(Lt)$, i.e., Lg is the Gödel number of $B(\boldsymbol{m})$. By the definition of $d(m, n)$,we have $d(m, Lg)$, hence $D(\boldsymbol{m}, \boldsymbol{Lg})$ is definable in $\boldsymbol{\mathcal{N}}$. However, recalling the logical structure of $B(\boldsymbol{m})$, which is a universally quantified conditional statement, we may infer $\neg T(\boldsymbol{Lg})$, i.e., Lt is not true in model X; hence, $\boldsymbol{Lg} \notin X$, which shows that X may only be null. In other words, since T has no model, the truth predicate function is arithmetically undefinable.

Acknowledgments: The authors thank Zhangju Liu for his insightful and useful discussions.

 doi:10.20944/preprints202412.2496.v1

References

1. Lang, H. and Liu, Z. (2020). A review of Lie 2-algebra. Advances in Mathematics (China), Vol.49, No. 6.
2. Leinster, T. (2017). Basic category theory. Cambridge University Press.
3. Lawvere, F. W. (1969). Diagonal Arguments and Cartesian Closed Categories. Lecture Notes in Mathematics, 92, 134-145.
4. Margaris, A. (1967). First order mathematical logic. Blaisdell Publishing Company.
5. Xiong M. (2016). Tarski's theorem and truth-theoretic paradoxes. Science Press, Beijing.
6. Yang, Y. (2020). Logical foundations of local gauge symmetry and symmetry breaking.
7. Cornell arXiv13582. Online at arXiv:2004.13582 [math.LO].