Incompleteness in a General Setting

John L. Bell

Full proofs of the Gödel incompleteness theorems are highly intricate affairs. Much of the intricacy lies in the details of setting up and checking the properties of a coding system representing the syntax of an object language (typically, that of arithmetic) within that same language. These details are seldom illuminating and tend to obscure the core of the argument. For this reason a number of efforts have been made to present the essentials of the proofs of Gödel's theorems without getting mired in syntactic or computational details. One of the most important of these efforts was made by Löb [8] in connection with his analysis of sentences asserting their own provability. Löb formulated three conditions (now known as the *Hilbert-Bernays-Löb* derivability conditions), on the provability predicate in a formal system which are jointly sufficient to yield the Gödel's second incompleteness theorem for it. A key role in Löb's analysis is played by (a special case of) what later became known as the *diagonalization* or *fixed point* property of formal systems, a property which had already, in essence, been exploited by Gödel in his original proofs of the incompleteness theorems. The fixed point property plays a central role in Lawvere's [7] category-theoretic account of incompleteness phenomena (see also [10]). Incompleteness theorems have also been subjected to intensive investigation within the framework of *modal logic* (see, e.g.[4], [5]). In this formulation the modal operator takes up the role previously played by the provability predicate, and the derivability conditions on the latter are translated into algebraic conditions (the so-called *GL,* i.e., Gödel–Löb, conditions) on the former.

My purpose here is to present a framework for incompleteness phenomena, fully compatible with intuitionistic or constructive principles, in which the idea of a coding system is *retained*, only in a simple, but very general form, a form wholly free of syntactical notions. As codes we shall take the elements of an arbitrary given nonempty set, possibly, but not necessarily, the set of natural numbers. As the objects to be encoded we take the elements of a second arbitrary nonempty set called the set of *sentences*: these are the counterpart of the sentences of a given formal language. We shall also suppose that the set of sentences is equipped with an equivalence relation which corresponds to the relation of provable equivalence with respect to a theory. Equivalence classes with respect to this equivalence relation will be called *propositions.*[1]

We shall take as our background theory intuitionistic set theory in any of its usual formulations (e.g. that presented in [1])[2]. We assume given two sets: Σ, the set of *sentences*, and C, the set of *codes*: we also assume that both Σ and C contain at least one element. The elements of the exponential[3] Σ^C may be considered as corresponding to formulas with one free variable ranging over C. Constant elements of Σ^C may be identified with sentences. For $\sigma \in \Sigma$ we write $\tilde{\sigma}$ for the map $C \to \Sigma$ with constant value σ.

We also assume given:

---

[1] We note that for classical theories propositions in the above sense form a Boolean algebra, but for the intuitionistic theories which we shall have in mind propositions constitute a *Heyting* algebra, that is, a (distributive) lattice $(L, \leq, \wedge, \vee)$ with top and bottom elements 1, 0 equipped with a binary operation $\Rightarrow$ satisfying $x \wedge y \leq z$ iff $x \leq y \Rightarrow z$. We define the operations $\neg$ and $\Leftrightarrow$ by $\neg x = x \Rightarrow 0$ and $x \Leftrightarrow y = (x \Rightarrow y) \wedge (y \Rightarrow x)$. When the elements of a Heyting algebra are regarded as propositions arising from a theory, the relation $\leq$ represents entailment and the operations $\wedge$, $\vee$, $\Rightarrow$, $\neg$, $\Leftrightarrow$, 0, 1 represent conjunction, disjunction, implication, negation, bi-implication, and refutable and provable propositions, respectively. For a proposition a, the assertion that $a = 1$ expresses the condition that a is *provable.* The *consistency* of the theory is expressed by the assertion that $0 \neq 1$, that is , by the assertion that its corresponding algebra of propositions has at least two elements.

[2] For definiteness we could take our background theory to be Zermelo set theory formulated within intuitionistic first-order logic. For much of the development in the paper, one may take classical set theory as background theory and note that no use of the law of excluded middle is made.

[3] The exponential X^Y of two sets X, Y is the set of functions from X to $Y.$

- An equivalence relation $\approx$ on Σ. For $\sigma \in \Sigma$ we write $[\sigma]$ for the $\approx$-equivalence class of σ and $\Omega(\Sigma)$, or simply Ω, for the set of all such equivalence classes. The members of Ω are called *propositions*, and for $\sigma \in \Sigma$, $[\![\sigma]\!]$ is called the proposition *associated* with the sentence σ.
- A submonoid[4] L of Σ^Σ, each member of which preserves $\approx$. The members of L are called *logical maps*. Each logical map (more generally, any self-map on Σ preserving $\approx$) f then induces a map $f^*\colon \Omega \to \Omega$ given by $f^*([\![\sigma]\!]) = [\![f(\sigma)]\!]$ for $\sigma \in \Sigma$. We sometimes identify a logical map with the self-map on Ω it induces: thus logical maps may also be thought of as acting on Ω.
- A subset K of Σ^C containing each constant function $\tilde{\sigma}$. The members of K are called *coded functions*. We suppose that L acts on K, that is, K is closed under composition with members of L on the left.
- A *coding map* $k\colon K \to C$. We shall usually write $\ulcorner\varphi\urcorner$ for $k(\varphi)$. By abuse of notation we shall, for $\sigma \in \Sigma$, write simply $\ulcorner\sigma\urcorner$ for $\ulcorner\tilde{\sigma}\urcorner$. The elements $\ulcorner\varphi\urcorner$, $\ulcorner\sigma\urcorner$ are called the *codes* of φ and σ respectively.
- A *substitution map* $s\colon C \times C \to C$ satisfying $s(\ulcorner\varphi\urcorner, u) = \ulcorner\varphi u\urcorner$ for $\varphi \in K$, $u \in C$.

We define the associated *diagonal map* $d\colon C \to C$ by

$$d(u) = s(u, u) \quad \text{for } u \in C.$$

Clearly d satisfies

$$d(\ulcorner\varphi\urcorner) = \ulcorner\varphi(\ulcorner\varphi\urcorner)\urcorner \quad \text{for } \varphi \in K.$$

---

[4] That is, a subset of Σ^Σ containing the identity map on Σ and closed under composition.

We shall assume that K is closed under composition with d on the right: if $\varphi \in K$, then $\varphi \circ d \in K$.

We shall call a septuple $\mathscr{A} = (\Sigma, C, \approx, K, L, k, s)$ subject to the above data a *coding assemblage.* If $\Omega(\Sigma)$ has at least two elements, $\mathscr{A}$ is said to be *consistent.*

We now assume a fixed coding assemblage to be given.

A self-map f on Σ is *codable* if it preserves $\approx$ and, for some map $\varphi \in K$, we have $[\![f(\sigma)]\!] = [\![\varphi(\ulcorner\sigma\urcorner)]\!]$ for all $\sigma \in \Sigma$. In this sense f can be represented as a (coded) function of codes. Such a map φ is called a *coding representation* for f. Given $\varphi \in K$, the map $\varphi^\dagger\colon \sigma \mapsto \varphi(\ulcorner\sigma\urcorner)\colon \Sigma \to \Sigma$ is called the self-map on Σ *induced by* φ; if $\varphi^\dagger$ preserves $\approx$, we shall say that φ is *equable.* Clearly, if φ is equable, $\varphi^\dagger$ is then codable.

Self-maps on Ω induced by codable self-maps on Σ are called *codable self-maps on* Ω.

We can now prove

Proposition 1. The following conditions are equivalent:

(i) every logical map on Σ is codable;

(ii) the identity map 1_Σ on Σ is codable.

Proof. (i) $\Rightarrow$ (ii) is obvious. For the converse, the assumption gives a map $\tau \in K$ for which $[\![\sigma]\!] = [\![\tau(\ulcorner\sigma\urcorner)]\!]$ for all $\sigma \in \Sigma$. Clearly, for any logical map f, the map $f \circ \tau$ is a coding representation for f. ■

A coding representation τ for 1_Σ is called a *Tarski map*: it satisfies

$$[\![\sigma]\!] = [\![\tau(\ulcorner\sigma\urcorner)]\!].$$

This is the counterpart in our framework of what is called in the literature *Tarski's T-scheme*. A Tarski map therefore corresponds to a *truth definition.* From the equation above we see that a Tarski map, or truth definition, is just a *left inverse* up to ≈-equivalence for the coding map $\sigma \mapsto \ulcorner\sigma\urcorner: \Sigma \to C$.

Our next task is to prove a version of what is usually known as the Diagonalization Lemma ([9], Theorem 2.2.1). Given a (≈-preserving) self-map f on Σ, let us call a ⟦fixed⟧ point of f any element $\alpha \in \Sigma$ for which $[\![f(\alpha)]\!] = [\![\alpha]\!]$. Clearly, α is a ⟦fixed⟧ point of f if and only if $[\![\alpha]\!]$ is a fixed point in the usual sense for the self-map on Ω induced by f. Our version of diagonalization is then the

⟦Fixed⟧ Point Lemma. Any codable self-map on Σ has a [fixed] point.

Proof. Suppose $f: \Sigma \to \Sigma$ is codable with coding representation $\varphi: C \to \Sigma$. Let $c = \ulcorner\varphi \circ d\urcorner$ and put $\alpha = \varphi(d(c))$. Then

$$\ulcorner\alpha\urcorner = \ulcorner\varphi(d(c))\urcorner = \ulcorner(\varphi \circ d)(\ulcorner\varphi \circ d\urcorner)\urcorner = d(\ulcorner\varphi \circ d\urcorner) = d(c).$$

Hence

$$[\![f(\alpha)]\!] = [\![\varphi(\ulcorner\alpha\urcorner)]\!] = [\![\varphi(d(c))]\!] = [\![\alpha]\!],$$

so that α is a ⟦fixed⟧ point for f. ■

This my be restated as the

Fixed Point Lemma. Any codable self-map on Ω has a fixed point.

It now follows from Proposition 1 and the ⟦Fixed⟧ Point Lemma that if a coding assemblage has a Tarski map, then every logical map on Σ has a ⟦fixed⟧ point. This immediately yields

Tarski's Theorem. If the set Σ of sentences in a coding assemblage $\mathscr{A}$ has a logical map with no ⟦fixed⟧ points, or equivalently, if the set Ω has a logical map with no fixed points, then $\mathscr{A}$ has no Tarski map. ■

Tarski's theorem in this formulation applies in particular when Ω is a Heyting algebra with at least two elements, and the negation map ¬ on Ω—which then has no fixed points—is included among the logical maps. This in turn enables us to recapture the usual formulation of Tarski's theorem on the undefinability of truth. For the algebra of propositions of a consistent theory *T* has at least two elements, so, provided the language of *T* meets the modest requirements for generating a coding assemblage (along the lines of example 1 immediately below), it follows that the associated coding assemblage has no Tarski map. This means that the language for *T* contains no truth definition for *T*; in a word, truth for *T* is *undefinable* in *T.*

Examples

1. Peano arithmetic. In this case the ingredients of the coding assemblage $\mathscr{P}$ —the *Peano assemblage* —are as follows: Σ is the set of sentences of the language $\mathscr{L}$ of first-order intuitionistic arithmetic **P**, *C* is the set *N* of natural numbers, ≈ is the relation of provable equivalence from **P**[5] *K* is the set of maps of the form φ: $n \mapsto \psi(\mathbf{n})$ where $\psi(x)$ is a

[5] Thus Ω may be regarded as the set of sentences of $\mathscr{L}$ identified up to provable equivalence from **P.**

formula of $\mathscr{L}$ with at most one free variable and **n** is the term of $\mathscr{L}$ representing *n*. The coding map *k* is given by $k(\varphi) = \#\psi$ where # is any standard Gödel numbering of the formulas of $\mathscr{L}$. The map s: $N \times N \to N$ is a recursive substitution function on Gödel numbers[6]. Finally L consists of the maps $\sigma \mapsto \sigma$, $\sigma \mapsto \neg\sigma$, $\sigma \mapsto \neg\neg\sigma$.

Assuming that **P** is consistent, it follows, as observed above, that $\mathscr{P}$ has no Tarski map, and so truth in $\mathscr{P}$ is undefinable in $\mathscr{P}$.

2. Intuitionistic set theory. Just as in classical set theory the power set PA of any set A is a Boolean algebra under the usual set-theoretic operations, so in intuitionistic set theory the power set is, under the same operations, a Heyting algebra. In particular, writing 1 for the one-element set {0}, P1 is a Heyting algebra which we shall denote by $\prod$. If σ is a sentence of the language of set theory, we write $\{0\,|\,\sigma\}$ for the element $\{x\colon x = 0 \wedge \sigma\}$ of $\prod$. From the axiom of extensionality it follows that $\{0\,|\,\sigma\} = \{0\,|\,\sigma'\}$ iff $\sigma \leftrightarrow \sigma'$. Thus the elements of $\prod$ correspond naturally to what we have termed *propositions,* in this case, to sentences identified under provable equivalence from the axioms of intuitionistic set theory. Under this correspondence each element $\omega \in \prod$ is correlated with the proposition $0 \in \omega$, and each proposition σ with the element $\{0\,|\,\sigma\}$ of $\prod$.

$\prod$ also plays the role of a *subset classifier.* That is, for each set A, subsets of A are correlated bijectively with maps $A \to \prod$: each subset $X \subseteq A$ is correlated with the map $x \mapsto \{0\,|\,x \in X\} : A \to \prod$, and each map $f : A \to \prod$ with the subset $f^{-1}(1)$ of A. The top element 1 (bottom element $\varnothing$) 1 of $\prod$ is identified with the true (false) proposition(s). In this way $\prod^A$ is seen to be naturally isomorphic to PA.

[6] See, e.g. [3], Example 7.4.5.

Now let us attempt to build a coding assemblage using $\prod$ as the underlying set of sentences and the identity relation as the underlying equivalence relation. Here it is natural to take C, the set of codes, to be any set containing at least one element, and to take $L = \prod^{\prod}$ and $K = \prod^{C}$. Using the observation immediately above, we may then identify K with PC. Take the coding map k to be an arbitrary map P$C \to C$; for $X \in$ PC, write $\ulcorner X \urcorner$ for $k(X)$.

For $\omega \in \prod$, the constant map $\tilde{\omega}: C \to \prod$ is correlated with the element $\omega^* = \{x \in C\colon 0 \in \omega\}$ of PC; it will be convenient to write $\ulcorner\omega\urcorner$ for $\ulcorner\omega^*\urcorner$. The map $\omega \mapsto \ulcorner\omega\urcorner\colon \prod \to C$ is the *coding map* on $\prod$.

The sextuple $\mathscr{Q} = (\prod, C, =, \mathrm{P}C, \prod^{\prod}, k)$ comprises six of the seven ingredients required to specify a coding assemblage. Can $\mathscr{Q}$ itself be enlarged to a coding assemblage? As we shall see, this cannot be done when the coding map on $\prod$ satisfies the modest requirement of being injective[7].

For under the assumption that $\ulcorner\bullet\urcorner\colon \prod \to C$ is injective, it is easy to check that it has a left inverse τ given by

$$\tau(x) = \bigcup\{X \subseteq 1\colon \ulcorner X \urcorner = x\}.$$

So if $\mathscr{Q}$ could be enlarged to a coding assemblage, Tarski's theorem would be violated, since $\prod$ evidently has at least two distinct elements.

In fact, if $\ulcorner\bullet\urcorner$ is injective, an argument akin to that for Russell's paradox shows directly that $\mathscr{Q}$ cannot even be augmented by a diagonal map. For any such diagonal map d would then have to satisfy

$$(*) \qquad d(\ulcorner X \urcorner) = \ulcorner\{0 \mid \ulcorner X \urcorner \in X\}\urcorner$$

[7] The modesty of this requirement is more easily seen when the background theory is classical (i.e. the law of excluded middle holds). For then $\Omega = \{\varnothing, 1\}$ and injectivity of $\ulcorner\bullet\urcorner$ boils down simply to $\ulcorner\varnothing\urcorner \neq \ulcorner 1 \urcorner$: that is, the true and the false receive different codes.

for $X \in \mathrm{P}C$. Now define

$$U = \{x \in C\colon\ d(x) = \ulcorner \varnothing \urcorner\}.$$

Then, using (*) and the injectivity of $\ulcorner \bullet \urcorner$,

$$\begin{aligned} \ulcorner U \urcorner \in U &\leftrightarrow\ d(U) = \ulcorner \varnothing \urcorner \\ &\leftrightarrow\ \ulcorner \{0 \mid \ulcorner U \urcorner \in U\} \urcorner = \ulcorner \varnothing \urcorner \\ &\leftrightarrow\ \{0 \mid \ulcorner U \urcorner \in U\} = \varnothing \\ &\leftrightarrow\ \ulcorner U \urcorner \notin U \end{aligned}$$

and we have a contradiction.

We now turn to Gödel's theorems.

We have seen that every codable self-map g on Σ has a [fixed] point. Let us call an element $\alpha \in \Sigma$ a *strong* ⟦fixed⟧ point for g if, for all $\sigma \in \Sigma$, we have

(*) $$[\![g(\sigma)]\!] = [\![\alpha]\!] \leftrightarrow [\![\sigma]\!] = [\![\alpha]\!].$$

We next prove another version of the Diagonalization Lemma, namely, the

Distinguished Element Lemma. Suppose that Σ has a codable self-map g with a strong ⟦fixed⟧ point α. Then, for any logical map f on Σ there is $\beta \in \Sigma$ such that

$$[\![f(\beta)]\!] = [\![\alpha]\!] \leftrightarrow [\![\beta]\!] = [\![\alpha]\!].$$

Proof. Let π be a coding representation for g. Given f: $\Omega \to \Omega$, define $c = \ulcorner f \circ \pi \circ d \urcorner$, $\beta = \pi(d(c))$. Then we have

$$\ulcorner f(\beta) \urcorner = \ulcorner f(\pi(d(c))) \urcorner = = \ulcorner (f \circ \pi \circ d)(\ulcorner f \circ \pi \circ d \urcorner) \urcorner = d(\ulcorner f \circ \pi \circ d \urcorner) = d(c).$$

Hence, using (*)

$$[\![f(\beta)]\!] = [\![\alpha]\!] \leftrightarrow [\![g(f(\beta))]\!] = [\![\alpha]\!] \leftrightarrow$$
$$\leftrightarrow [\![\pi(\ulcorner f(\beta) \urcorner)]\!] = [\![\alpha]\!] \leftrightarrow [\![\pi(d(c))]\!] = [\![\alpha]\!] \leftrightarrow [\![\beta]\!] = [\![\alpha]\!]. \quad \blacksquare$$

Now we can formulate Gödel's First Incompleteness Theorem in the present setting. Here we require Σ to have a distinguished element $\top$: we think of $\top$ as representing the *provable* sentences in the sense that $[\![\top]\!]$ is taken to be the equivalence class consisting of all provable sentences. We suppose given an equable map $\pi \in K$ which we shall term a *provability* map, in the sense that, for each $\sigma \in \Sigma$, the element $\pi(\ulcorner \sigma \urcorner)$ of Σ shall be construed as the sentence σ *is provable.* The self-map g on Σ induced by π is then necessarily codable. We shall call g a *Gödel map* if it has $\top$ as a strong [fixed] point. If g is a Gödel map, then the provability map π satisfies

$$[\![\sigma]\!] = [\![\top]\!] \leftrightarrow [\![\pi(\ulcorner \sigma \urcorner)]\!] = [\![\top]\!].$$

This may be construed as asserting that a sentence σ is provable iff the sentence $\pi(\ulcorner \sigma \urcorner)$ asserting the provability of σ is itself provable. Notice that the self-map on Σ induced by a Tarski map is a Gödel map.

Now call a coding assemblage *Gödelian* if it is consistent, has a Gödel map, and there is an element $\perp$ of Σ such that $[\![\top]\!] \neq [\![\perp]\!]$ together with a logical map ν on Σ such that $[\![\nu(\top)]\!] = [\![\perp]\!]$ and $[\![\nu(\perp)]\!] = [\![\top]\!]$.

We think of ν as the *negation* operation on sentences and $\perp$ as representing the *refutable* sentences.

We can now prove

Gödel's First Incompleteness Theorem. The set of propositions of any Gödelian coding assemblage has at least three elements.

Proof. Given a Gödelian coding assemblage there is, by the Distinguished Element Theorem, an element $\beta \in \Sigma$ for which $[\![\nu(\beta)]\!] = [\![\top]\!] \leftrightarrow [\![\beta]\!] = [\top]$. In that case $[\![\top]\!] \neq [\![\beta]\!] \neq [\![\bot]\!]$, so that Ω has the three distinct elements, $[\![\top]\!]$, $[\![\bot]\!]$, $[\![\beta]\!]$. ■

An element $\beta \in \Sigma$ such that $[\![\top]\!] \neq [\![\beta]\!] \neq [\![\bot]\!]$ evidently represents an *undecidable* sentence, so the theorem just proved may be taken to assert that *any Gödelian coding assemblage contains undecidable sentences.*

All this applies in particular to the Peano coding assemblage $\mathscr{P}$. Let $\top$ be the sentence $\mathbf{0} = \mathbf{0}$, and $\bot$ the sentence $\mathbf{0} = \mathbf{1}$. Also let *Prov* be a provability predicate for **P.** Then, by standard arguments[8], we have, for any arithmetical sentences α, β,

$$(\mathbf{Prov_1}) \qquad \vdash_{\mathbf{P}} \alpha \leftrightarrow \vdash_{\mathbf{P}} Prov(\#\alpha)$$

$$(\mathbf{Prov_2}) \qquad \vdash_{\mathbf{P}} Prov(\#(\alpha \rightarrow \beta)) \rightarrow [Prov(\#\alpha) \rightarrow Prov(\#\beta)])$$

$$(\mathbf{Prov_3}) \qquad \vdash_{\mathbf{P}} Prov(\#\alpha) \rightarrow Prov(Prov(\#\alpha)).$$

Now let π: $N \rightarrow \Sigma$ be the map $n \mapsto Prov(\mathbf{n})$. It follows from ($\mathbf{Prov_2}$) that π is equable, and from ($\mathbf{Prov_1}$) that the map g: $\Sigma \rightarrow \Sigma$ induced by π has $\top$ as a strong $[\![$fixed$]\!]$ point. Accordingly g is is a Gödel map for $\mathscr{P}$. Assuming

[8] See, e.g. [6], Ch. 16.

that **P** is consistent, $\mathscr{P}$ is then Gödelian, and accordingly contains undecidable propositions.

Finally let us set about formulating Gödel's Second Incompleteness Theorem in the present setting. To do this we need to introduce the concept of a Hilbert-Bernays-Löb, or HBL-operator. Let us assume that $\mathscr{A}$ is a coding assemblage in which Ω is a Heyting algebra. An *HBL-operator* in $\mathscr{A}$ is a self-map $\Box$ on Ω satisfying the conditions:

(a) $\Box 1 = 1$

(b) $\Box(x \Rightarrow y) \leq (\Box x \Rightarrow \Box y)$

(c) $\Box x \leq \Box\Box x$.

An HBL-operator may be considered a modal operator satisfying the K4 axioms[9]. It follows immediately from (a) and (b) that $\Box$ is order-preserving.

We may think of $\Box$ as a provability operator acting on propositions: for each proposition x, $\Box x$ is the proposition asserting "x is provable". In that case (a) above asserts: *if x is a provable proposition, then so is the proposition "x is provable*"; (b) asserts: *the proposition "x implies y is provable" implies the proposition '"x is provable" implies "y is provable*"'; and (c) asserts: *the proposition "x is provable" implies the proposition "'x is provable' is provable".*

Now let us call a coding assemblage $\mathscr{A}$ *suitable*[10] if (i) Ω is a Heyting algebra with an HBL operator and (ii) for each $a \in \Omega$ the map $x \mapsto (\Box x \Rightarrow a)\colon \Omega \to \Omega$ is codable.

We can now prove a version of

[9] See [4], p. 5.
[10] I.e., suitable for proving Gödel's second incompleteness theorem: see below.

Löb's Theorem[11]**.** Let $\mathscr{A}$ be an suitable coding assemblage with HBL-operator $\Box$. Then, for any $a \in \Omega$

(i) $\Box(\Box a \Rightarrow a) \leq \Box a.$

(ii) $\Box a \leq a \;\rightarrow\; a = 1.$

Proof. Given $a \in \Omega$, the map $x \mapsto (\Box x \Rightarrow a)\colon \Omega \rightarrow \Omega$ is codable and so by the Fixed Point Lemma has a fixed point b. That is,

$$(*) \qquad b = (\Box b \Rightarrow a).$$

A fortiori $b \leq (\Box b \Rightarrow a)$, whence

$$\Box b \leq \Box(\Box b \Rightarrow a) \leq (\Box\Box b \Rightarrow \Box b).$$

Hence

$$(**) \qquad \Box b = \Box b \wedge \Box\Box b \leq \Box a$$

It follows that $(\Box a \Rightarrow a) \leq (\Box b \Rightarrow a)$, whence

$$\begin{aligned} \Box(\Box a \Rightarrow a) &\leq \Box(\Box b \Rightarrow a) \\ &= \Box b \;\; \text{(by (*))} \\ &\leq \Box a \;\; \text{(by (**))}. \end{aligned}$$

This gives (i).

For (ii), we assume $\Box a \leq a$, so that $(\Box a \Rightarrow a) = 1$. It now follows from (i) that

$$1 = \Box 1 = \Box(\Box a \Rightarrow a) \leq \Box a.$$

[11] Theorem 4.1.1 of [6].

Therefore $\Box a = 1$, and since $\Box a \leq a$, we conclude that $a = 1$. ■

Remarks. (1) From (i) of Löb's Theorem we see that $\Box$ satisfies the so-called *GL* (Gödel-Löb) *axiom*[12] for a normal modal logic, i.e. the scheme

$$\Box(\Box A \to A) \to \Box A.$$

(2) If $\Box$ is the identity map, then we deduce from (ii) of Löb's Theorem that Ω has just one element, that is, $\mathscr{A}$ is inconsistent.

From Löb's Theorem one derives:

Gödel's Second Incompleteness Theorem. Given a suitable consistent coding assemblage $\mathscr{A}$ with HBL operator $\Box$. Then, for any $x \in \Omega$, $\Box x \neq 0$; or equivalently, $\neg\Box x \neq 1$. In particular, $\Box 0 \neq 0$; or equivalently $\neg\Box 0 \neq 1$.

Proof. If, $\Box x = 0$; then $\Box 0 \leq \Box x = 0$; hence by Löb's theorem $0 = 1$, and it follows that $\mathscr{A}$ is inconsistent. ■

Consider again the Peano assemblage $\mathscr{P}$. There Ω is a Heyting algebra and conditions **Prov$_{1\text{-}3}$** on the provability predicate imply that the self-map $\Box$ on Ω induced by the Gödel map g is an HBL-operator. It is also easily checked for each $a \in \Omega$ the map $x \mapsto (\Box x \Rightarrow a): \Omega \to \Omega$ is codable. Accordingly $\mathscr{P}$ is suitable, and so the 2nd incompleteness theorem applies to it.

Thinking of $\Box$ as a provability operator, $\neg\Box x$ is the proposition "*x is unprovable*", so that $\neg\Box x = 1$ may be taken as asserting the provability of "*x is unprovable*". In that case the second incompleteness theorem may be taken to assert that in any suitable consistent coding assemblage,

---

[12] See [5], p. 5.

there is no proposition whose unprovability is provable. In particular, the unprovability of the proposition 0 is also unprovable. Now the unprovability of 0—that is, of any refutable proposition—may be considered a form of consistency: let us call it *internal consistency.* This terminology enables the second incompleteness theorem as stated above to assume a more familiar form: in any suitable consistent coding assemblage, *its internal consistency is unprovable.* This applies in particular to the Peano assemblage.

On the other hand, observe that the assertion $\Box 0 = 1$ represents what is naturally termed *internal inconsistency.* By taking the HBL-operator $\Box$ to be identically 1, in other words, taking every proposition to satisfy the internal condition "*x is provable*", so that, in particular, $\Box 0 = 1$, we see that consistency, that is, the condition that Ω has two distinct elements, is compatible with internal inconsistency. This just shows that internal consistency need have little to do with consistency, or, more generally, that provability maps need have little to do with provability[13].

In conclusion, we point out that while in stating and proving these results we have used ordinary set-theoretic language, they can be formulated in toposes (see, e.g. [2]) or more general categories (cf. the discussion in [10]).

References

[1] Bell, John L. *Set Theory: Boolean-Valued Models and Independence Proofs.* Clarendon Press, Oxford, 2005.

[2] Bell, John L. *Toposes and Local Set Theories.* Clarendon Press, Oxford, 1988.

[3] Bell, John L., and Machover, M. *A Course in Mathematical Logic.* North-Holland, 1977.

[13] An observation also made in [6].

[4] Boolos, G. *Gödel's second incompleteness theorem explained in words of one syllable.* Mind 103 (1994), no. 409, 1-3.

[5] Boolos, G. *The Logic of Provability.* Cambridge University Press 1995.

[6] Boolos, G. and Jeffrey, R. *Computability and Logic.* Cambridge University Press, 1974.

[7] Lawvere, F.W. *Diagonal arguments and cartesian closed categories.* Category theory, homology theory and their applications, II (Battelle Institute Conference, Seattle, Wash., 1968). Springer, Berlin, 1969, 134-145.

[8] Löb, M. H. *Solution of a problem of Leon Henkin.* Journal of Symbolic Logic 20 (1955), 115-18.

[9] Smorynski, C. *The Incompleteness Theorems.* Handbook of Mathematical Logic, J. Barwise, ed., North-Holland, 1977, pp. 821-866.

[10] Yanofsky, N. *A universal approach to self-referential paradoxes, incompleteness and fixed points.* Bull. Symb. Logic 9(3), 2003, 362-386.

Department of Philosophy,

University of Western Ontario

e-mail: jbell@uwo.ca