The Liar Paradox and Metamathematics

Richard Kimberly Heck

November 10, 2023

Informal presentations of the liar paradox typically begin with a sentence like this one:

> This sentence is not true.

or perhaps:

(L) Sentence (L) is not true.

Such sentences are 'self-referential' in the straightforward sense that they refer to themselves, and then go on to say of themselves that they are not true. Reasoning whose specifics are explored in other chapters of this book[1] then leads to a paradox: If (L) is true, then, since what it says is precisely that it is not true, it is not true; but if it is not true, then, since what it says is that it is not true, it must be true after all. Hence, (L) can be neither true nor false; but then it is not true, and off we go again.

A common first reaction to this paradox is to blame the self-referential character of the sentences that give rise to it. One can easily observe this reaction among students and (patient) friends, and it surfaces sometimes in the older literature on truth, as well. J. L. Austin (1950, pp. 121–3), for example, argues that there is a general prohibition on self-reference. This response is largely dismissed nowadays, however, for two main reasons. First, a very similar construction, involving a sentence like

(G) Sentence (G) is not provable.

is used in the proof of Kurt Gödel's (1931) famous incompleteness theorems, and analogous constructions turn up elsewhere in mathematical logic. Second, the techniques that Gödel uses to construct sentences

[1] Note the self-reference!

like (G) require only very minimal resources. It is this that leads Saul Kripke (1975, p. 692) to remark that "Gödel put the issue of the legitimacy of self-referential sentences beyond doubt; he showed that they are as incontestably legitimate as arithmetic itself". As we shall see below, Kripke is, strictly speaking, correct, but the remark can be misleading, and it may well have misled some authors.

In contemporary presentations, either of Gödel's results or of the Liar Paradox, the self-referential construction is typically presented as a lemma: the so-called Diagonal Lemma.[2]

Lemma (Diagonal Lemma)**.** *Let $A(x)$ be an arbitrary formula containing just the variable x free, and let Σ be a 'sufficiently strong' formal theory. Then there is a sentence G_A such that Σ proves*

$$G_A \equiv A(\ulcorner G_A \urcorner)$$

The significance of the corner quotes, $\ulcorner \cdot \urcorner$, will be explained below; for now, it suffices to note that they are meant to function somewhat like ordinary quotation. We'll also discuss below exactly what it means for Σ to be 'sufficiently strong'; for now, it suffices to note that such familiar theories as Dedekind-Peano arithmetic (PA) and Zermelo–Fraenkel set theory (ZF) are sufficiently strong and, indeed, are much stronger than is required.

With the diagonal lemma in hand, then, we can then proceed as follows. Suppose we are working in some 'sufficiently strong' theory whose language includes a 'truth-predicate' $\mathsf{Tr}(x)$.[3] If we take as our formula $A(x)$ the formula $\neg\mathsf{Tr}(x)$, then the Diagonal Lemma delivers a sentence L such that the equivalence

$$L \equiv \neg\mathsf{Tr}(\ulcorner L \urcorner)$$

will be provable. There is thus at least a weak sense in which L 'says of itself' that it is not true: It is provably equivalent to the sentence $\neg\mathsf{Tr}(\ulcorner L \urcorner)$, which straightforwardly says that L is not true.[4] We'll see

[2] Why the 'Diagonal' Lemma? I do not know who first used this term, but there is a clear analogy to the sort of diagonal construction used in Cantor's proof of his eponymous theorem. The term "Fixed Point Lemma", which one also sees, has a more obvious genealogy: x is a fixed point of $f(\cdot)$ iff $x = f(x)$.

[3] So as not to clutter the exposition, I will usually omit quotation marks when citing formulae. Context should make clear enough what is intended.

[4] One might object that L will only 'say of itself' that it is not true if what's provable

below that there is a stronger sense in which L is self-referential, as well. But provable equivalence is adequate for most purposes.

In his original paper on the incompleteness theorems, Gödel himself does not formulate the Diagonal Lemma. Rather, he explicitly constructs a particular sentence, the famous "17 Gen r", which plays the role of (G) in his proof (Gödel, 1986, pp. 173ff).[5] Gödel does note that his proof of the incompleteness theorem, which specifically applies to a simplified version of the formal system of *Principia Mathematica* (Whitehead and Russell, 1925), which he calls P, generalizes to a wide class of theories (Gödel, 1986, p. 181); it is implicit in that remark that the construction of (G) can also be generalized. That, however, only involves making use of different notions of provability: provability in *Principia*, or provability in ZF, or provability in some other recursively axiomatized theory. It was Rudolf Carnap (1937, §35) who first realized that there was an even more general construction implicit in Gödel's work.[6] In retrospect, then, we can see that what Gödel did, in effect, was to take $A(x)$ to be $\neg\mathsf{Bew}_\Sigma(x)$, where $\mathsf{Bew}_\Sigma(x)$ is a 'provability predicate' for some recursively axiomatized[7] theory Σ, and then to construct a sentence G such that Σ itself will prove:

$$G \equiv \neg\mathsf{Bew}_\Sigma(\ulcorner G \urcorner)$$

What Carnap saw was that the construction of G is 'uniform' in the sense that none of its details depend upon our beginning with $\neg\mathsf{Bew}_\Sigma(x)$: Exactly the same construction can be applied starting with any other

in this theory is actually true, i.e., if the theory is sound. But it's enough if the fragment of the theory used in proving the equivalence is sound, and the axioms needed to prove the Diagonal Lemma are certainly true. See, however, Milne (2007) for some important observations along these lines.

[5] Gaifman (2006) offers a reconstruction of Gödel's discovery of the construction as a kind of synthesis of Cantor's construction and Richard's paradox of *the least number not nameable in fewer than eighteen syllables*. (There must be such a number, since there are only finitely many expressions of fewer than eighteen syllables. But the italicized phrase names it in just sixteen syllables.) Gaifman also discusses the relation between Gödel's construction and Kleene's proof of the recursion theorem and proves a general result that unifies these.

[6] Smoryński (1981), in an otherwise useful discussion of the history of the Diagonal Lemma, does not mention Carnap, suggesting that Rosser (1939, p. 57) was the first to state the Lemma in general form.

[7] A form of this restriction is present in Gödel's work, and in most expositions of the incompleteness theorems, but it is not actually required. The construction of the Gödel sentence goes through so long as the set of Σ's axioms is definable in Σ. This can be used to prove the incompleteness of such theories as PA plus all Π_1 truths.

formula.[8]

To understand the role that self-reference plays in the liar paradox, then, it will suffice to understand the Diagonal Lemma. Unfortunately, however, most presentations of that Lemma make it seem like magic: They do indeed show *that* self-referential sentences can be constructed, but they do not produce anything one might call 'understanding'.[9] The reason, I suggest, is that a proper statement of the Diagonal Lemma, let alone a proper proof of it, requires several different ingredients, each of which poses its own conceptual obstacle. In what follows, then, I want to pull these various ingredients apart. An element of magic will remain, but it's the kind of magic that's always present in really clever constructions.

Given that our goal here is understanding, then, I'm going to focus more on ideas than on the formal niceties. Readers who want more detail will find it in the papers I'll cite.[10]

1 The Diagonal Lemma in Theories of Syntax

Modern discussions of the liar paradox typically concern an arithmetical theory, such as PA, to which a truth-predicate $\mathsf{Tr}(x)$, or other semantic vocabulary, has been added. At first blush, this might seem very strange. Truth is a property of sentences,[11] but the language of PA does not seem

[8] As Gaifman (2006, p. 710) notes, this seems obvious now, but it would not have been at all obvious in the early 1930s.

[9] Which contributes, I suspect, to the persistent sense among non-specialists that there is something fishy about the incompleteness theorem.

[10] The standard textbook on this material is Boolos et al. (2007, esp. chs. 15–17). Another good reference is Smith (2013). Probably the most comprehensive study of these issues is Smullyan (1994), which was a kind of sequel to Smullyan (1982). Some years ago, I wrote a guide, for teaching purposes, that's intended to be accessible to students with just a good understanding of basic logic and some math skills (Heck, 2022). It skips the (difficult and lengthy) proof that all recursive functions are representable in Q but is otherwise reasonably rigorous.

[11] In natural languages, there are many bits of vocabulary that are 'context-dependent'. Standard examples include (in English) such words as "I", "here", and "now", whose reference, when uttered, depends upon when, where, and by whom they were uttered. The sentence "I am a philosopher", then, has no truth-value by itself; only particular utterances of it do. Exactly what we should regard as the primary bearers of truth—utterances, token sentences, propositions, etc—is itself a controversial matter. Here, however, we shall ignore this issue: In the sorts of formal languages we will be discussing, there is no context-dependence, so we can safely apply the truth-predicate to type sentences. Nonetheless, many discussions of truth suffer from an over-emphasis on

to provide us with any way to talk about sentences: In the language of arithmetic, it would seem, we can only talk about numbers; how, then, can we possibly say, in the language of PA, that the sentence "$0 = 0$" is true? As we shall see, there is a solution to this problem (due, again, to Gödel). But that is one of the ingredients we'll discuss later.

In this section, we'll begin by discussing theories of syntax: theories whose (explicit) subject matter is symbols and their properties. It's easy to describe a theory whose subject matter is its own syntax and so to which we could, if we wished, add a new 'semantic' predicate $\mathsf{Tr}(x)$ that applied to sentences of that very language. Before we get too enmeshed in technicalities, however, let us step away from formalisms and see how the construction of self-referential sentences works informally.

1.1 Self-Reference in Ordinary Language

Here's how to construct an expression that is a name of itself.[12]

We'll need to make use of the operation of substituting one expression for some part of another expression. We all know what this means. If E_1 is:

> Some expressions are x

and E_2 is the word:

> amusing

then the result of substituting E_2 for all occurrences of 'x' in E_1 is:

> Some expressions are amusing

Of course, there's no need for E_1 and E_2 to be different. So if E_2 is the same as E_1, then the result is:

> Some expressions are some expressions are x

which is nonsense, but a perfectly good expression (symbol, string).

A special case arises if E_2 is not just an expression but a *name* of an expression. So let E_1 be:

> x contains four words

formal languages, due to their lack of context-sensitivity. See Heck (2023).

[12] Smullyan (1994, §2) gives a similar exposition, though this kind of idea goes back at least to Quine (1981, §59), which was originally published in 1940. See also Smullyan (1957).

and let E_2 be the phrase:

> Junebug's favorite English sentence

Then the result of substituting E_2 for all occurrences of 'x' in E_1 is:

(1) Junebug's favorite English sentence contains four words

Whether this is true will depend upon what Junebug's favorite English sentence is. Then again, E_2 could be a quote-name of an expression, for example:

> "Junebug's favorite English sentence"

and now the result is:

(2) "Junebug's favorite English sentence" contains four words

which is certainly true. As always, it's critical to distinguish the case with quotes (mention) from the case without quotes (use). There's all the difference in the world between (1) and (2).

We could have E_2 be E_1 again, in which case we get nonsense:

> x contains four words contains four words

More interestingly, we can have E_2 be a quote-name of E_1. I.e., E_2 could be:

> "x contains four words"

and then the result is:

> "x contains four words" contains four words

which we might reasonably count as true.

Focus attention now on this special kind of self-substitution, that is, on the operation:

> The result of replacing all occurrences of 'x' in E_1 with its own quote-name

(that is, with a quote-name of E_1). The example we just considered is one case of this kind of self-substitution, with E_1 being:

> x contains four words

Other examples are easy to construct. E.g, if E_1 is:

> x contains one free variable

then the result is:

> "x contains one free variable" contains one free variable

which is true.

To get a self-referential expression, apply the operation just illustrated to an expression describing that very operation:

(3) The result of replacing all (unquoted) occurrences of 'x' in x with its own quote-name

The idea of doing this is what can't really be motivated: This is where the cleverness in the construction lies. But once one has hit upon this idea, the rest is easy. If we replace all (unquoted) occurrences of 'x' in (3) with a quote name of (3), we get:

(4) The result of replacing all (unquoted) occurrences of 'x' in "The result of replacing all (unquoted) occurrences of 'x' in x with its own quote-name" with its own quote-name

But now (4) describes the result of the very operation we just performed! So (4) is a name of itself.[13]

As said, the element of magic lies in the idea of applying the operation described by (3) to (3) itself. The other resources required are quite simple: substitution, and the formation of quote-names. Given an understanding of those, the 'proof' that (4) names itself is as simple as it could be: It just involves carrying out the substitution described in (4) and then observing that the result is (4) itself.

We can make this more concise by employing an abbreviation. Abbreviate (3) as: The diagonalization of x. Then consider:

(5) The diagonalization of "The diagonalization of x"

This expression too names itself.

[13] Those attracted to Russellian views of descriptions might want to object that (4) does not refer to anything, since descriptions are not referring expressions. We'll return to this sort of issue below.

1.2 The Diagonal Lemma, Informally

We can now prove the following informal version of the Diagonal Lemma. By a 'sentence-frame', let us mean something like "x is long" or "x contains five words": something that would be a sentence if we replaced all (unquoted) occurrences of x with (say) "Xander".[14]

Lemma. *Let $A(x)$ be a sentence-frame of English containing the variable x (but otherwise just ordinary words of English). Then there is an English expression t_A that is itself a name of the English sentence $A(t_A)$: the result of substituting the very term t_A for the variable x in $A(x)$. I.e., and roughly:*

$$t_A = \ulcorner A(t_A) \urcorner$$

The sentence $A(t_A)$ will thus be a sentence that 'says of itself' that it has whatever property $A(x)$ expresses.

We'll just do an example, but it will be clear that it generalizes. Let $A(x)$ be:

(6) x is weird

To construct an expression that names the result of substituting that very expression for 'x' in (6), just consider:[15]

(7) The result of replacing all occurrences of 'x' in x with its own quote-name is weird

And now ask: What is the result of applying the operation mentioned in the subject of (7) to (7) itself? That is, we want to know which expression is named by:

(8) The result of replacing all occurrences of 'x' in "The result of replacing all occurrences of 'x' in x with its own quote-name is weird" with its own quote-name

So just take (7) and replace the (unquoted) occurrence of 'x' in it with the result of putting (7) in quotes:

(9) The result of replacing all occurrences of 'x' in "The result of replacing all occurrences of 'x' in x with its own quote-name is weird" with its own quote-name is weird

[14] If we wanted to make things seem even less formal, we could use blanks and talk about sentence-frames like "_____ is tall".

[15] What I've done is to replace 'x' in (6) with (3). I'll henceforth omit the qualification "(unquoted)".

So (8) is a name of (9). We have proven this by a simple calculation. But (9) is just (8) followed by the words "is weird", i.e., it is the result of replacing 'x' in (6) with (8). Roughly:

> (8) = "(8) is weird"

So (8) says of itself that it is weird. Which it is.

Corollary. *Let $A(x)$ be a sentence-frame. Then there is a sentence G_A that 'says of itself' that it has whatever property $A(x)$ expresses. I.e., and roughly:*

$$G_A \equiv A(\ulcorner G_A \urcorner)$$

Proof. By the Diagonal Lemma, there is a term t_A such that $t_A = \ulcorner A(t_A)\urcorner$. By Leibniz's Law, then

$$A(t_A) \equiv A(\ulcorner A(t_A)\urcorner)$$

So the promised sentence G_A is $A(t_A)$. □

Note that proving this equivalence requires us to apply Leibniz's Law to the sentence-frame $A(x)$, that is, to make an inference of the form:

$$t = u \rightarrow A(t) \equiv A(u)$$

We'll consider the significance of this point below.

Using the abbreviation defined above, we can again make this more concise. Consider:

(10) The diagonalization of "The diagonalization of x is weird"

and note that it is a name of:

(11) The diagonalization of "The diagonalization of x is weird" is weird

So the subject of (11)—i.e., (10)—names (11). So (11) 'says of itself' that it is weird.

1.3 Self-Reference in Formal Theories of Syntax

As we have seen, the construction of a 'diagonal sentence' from any given formula $A(x)$ requires only very modest resources: Substitution and the formation of quote names, as well as the general ability to talk about expressions of the language in question. For the time being, I'll use corner quotes for this latter purpose, treating e.g. $\ulcorner\exists x\urcorner$ as a primitive, unstructured name of the contained string. I'll write $\mathsf{subst}(y, z)$ to mean: the result of substituting z for all free occurences of "x" in y; and write $\mathsf{q}(y)$ to mean: the quote name of y, a quote name being a primitive term like $\ulcorner\exists x\urcorner$.[16] Then we can define $\mathsf{diag}(y)$—the diagonalization of y—as: $\mathsf{subst}(y, \mathsf{q}(y))$, officially treating this as an abbreviation.

Given a formula $A(x)$, we can then consider

(12) $\mathsf{diag}(\mathsf{subst}(\ulcorner A(x)\urcorner, \ulcorner\mathsf{diag}(x)\urcorner))$

Or, more concisely, but slightly less precisely:

(13) $\mathsf{diag}(\ulcorner A(\mathsf{diag}(x))\urcorner)$

We can then observe that the result of carrying out the mentioned operation is

(14) $A(\mathsf{diag}(\ulcorner A(\mathsf{diag}(x))\urcorner))$

So (13) denotes (14). But the argument in (14) is (13), so (14) once again 'says of itself' that it has whatever property $A(x)$ expresses.

To formalize this construction, then, we need only to operate in a theory that allows us to define the relevant notions. As we'll see below, formalizing quotation raises some annoying but resolvable technical problems. But if we set these aside for the moment, then we can already see one straightforward way to formalize the proof of the diagonal lemma. Start with some language $\mathcal{L}$; add to it symbols $\mathsf{subst}(y, z)$, $\mathsf{q}(y)$, and primitive terms like $\ulcorner\exists x\urcorner$; take as axioms all true sentences of the form:

$$\mathsf{subst}(t, u) = v$$
$$\mathsf{q}(t) = v$$

where t, u, and v are quote names. These axioms will be sufficient to tell us which expression $\mathsf{diag}(\mathsf{subst}(\ulcorner A(x)\urcorner, \ulcorner\mathsf{diag}(x)\urcorner))$ is, in each case, and that is enough to prove the Diagonal Lemma.[17]

One might nonetheless balk at treating such a complex operation as substitution as primitive, so the question arises how it might be defined. This question was almost answered by Gödel, whose construction of a

[16] So, e.g., $\mathsf{q}(\ulcorner\exists x\urcorner) = \ulcorner\ulcorner\exists x\urcorner\urcorner$. I.e, the quote name of the string consisting of an existential quantifier followed by an ex is a string consisting of a left corner, followed by an existential quantifier, followed by an ex, followed by a right corner. (It's embedded quotation of this sort that causes the problems with quotation that we'll discuss below.)

[17] This may seem like cheating. But such a theory is not unlike the arithmetical

self-referential sentence also uses the operation of substitution, which he defines explicitly (Gödel, 1986, p. 167).[18] If one traces the steps of this definition, one can see that Gödel defines substitution in terms of concatenation: the operation of 'gluing' two strings together.[19] The hitch is that Gödel's definition uses (primitive) recursion extensively, and he does not show us how to effect primitive recursion using just concatenation; he does that, rather, using a theory of (finite) sequences based upon prime factorization.[20] However, if we *could* effect primitive recursion using just concatenation, then we could adapt Gödel's construction to define substitution in terms of it alone.

As it happens, this can in fact be done. Exactly how it is done depends upon how strong a theory of concatenation one employs. On one end of the spectrum, John Corcoran, William Frank, and Michael Maloney (1974) employ a second-order theory of concatenation, one that is comparable to second-order arithmetic. In that sort of theory, recursive definitions can be formalized using well-known techniques due to Richard Dedekind (1902) and Gottlob Frege (1879). Many years earlier, W. V. O. Quine (1946) had shown how to develop a theory of sequences within a theory of concatenation[21] and then to use it to 'construct' arithmetic

theory known as R, which has among its axioms all true equations of the forms:

$$\overline{n} + \overline{m} = \overline{k}$$
$$\overline{n} \times \overline{m} = \overline{k}$$

where $\overline{n}$ is the numeral for n. (We'll discuss R further below.) The crucial point, of course, is that substitution and the formation of quote names are algorithmic operations, so the theory mentioned in the text is recursively axiomatizable.

[18] These definitions come in three stages. First, Gödel (1986, 163–71) simply gives standard mathematical definitions of the relevant notions. Then, in Theorem V, he proves that these defintions can be formalized in the theory P (Gödel, 1986, p. 171). Later in the paper, he proves that these defintions are arithmetical, in the now standard sense. That is Theorem VII (Gödel, 1986, p. 183).

[19] So, for example, the concatenation of "ABC" and "123" is "ABC123".

[20] In fact, what get concatenated in Gödel's treatment are sequences, with strings treated as sequences of symbols.

[21] One might wonder what the difference is between concatenation and sequences. Aren't sequences, in effect, just strings of their elements? No, as one can see by considering the two sequences: <ab, cd, ef> and <abc, def>. The obvious idea, then, is to use some kind of separator: We don't just concatenate the elements of the sequence, but separate them with, say, a comma. But what if the elements of the sequence contain the comma? If we want to be able to construct all possible sequences of symbols, then we can't just ignore this problem. But this is precisely the problem Quine shows us how to solve. See note 42.

itself. Quine claims that his approach is "elementary", but he appears to mean by this that it is first-order. He does not specify any formal theory of concatenation in which he proposes to work and appears, in fact, to have in mind the theory consisting of *all truths* of the first-order theory of concatenation (and, correspondingly, of arithmetic).[22] It seems plausible, however, that Quine's construction can be carried out in a theory of concatenation analogous to PA.[23]

More recently, Andrzej Grzegorczyk (2005) showed how a construction much like Quine's can be carried out in an extremely weak theory of concatenation, which he calls TC.[24] Where $\frown$ symbolizes concatenation,[25] the axioms of TC are:

1. $(x \frown y) \frown z = x \frown (y \frown z)$
2. $x \frown y = z \frown w \rightarrow (x = z \land y = w) \lor$
   $\exists u[(x \frown u = z \land y = u \frown w) \lor (x = z \frown u \land u \frown y = w)]$
3. $\alpha \neq x \frown y$
4. $\beta \neq x \frown y$
5. $\alpha \neq \beta$

The core of the theory is really the first two axioms. The first, of course, is associativity; the second axiom is known as 'Tarski's law' or the 'editor axiom'.[26] Axioms (3)–(5) guarantee that there are at least two atoms, α and β. (Without them, the theory has a one-element model.) But, in general, a syntactic theory $\mathsf{TC}_{\mathcal{L}}$ for a language $\mathcal{L}$ will include such

[22] This is an incredibly powerful theory, strong enough to prove the consistency of every consistent formal theory, and so much stronger even then second-order arithmetic, or ZFC, or ZFC plus whatever large cardinals you like.

[23] This would be a theory that added an induction principle to Grzegorczyk's TC (to be mentioned momentarily). Or, probably equivalently, the theory Halbach and Leigh (2022, Ch. 8) call E^*.

[24] See Halbach and Leigh (2022, Ch. 5) for a different but also very weak theory in which the Diagonal Lemma can be proven.

[25] The other symbol commonly used is $*$, which is what Gödel uses for the operation of concatenating two sequences.

[26] Here's what this says. Suppose I split some string in the middle, with the 'head' being x and the 'tail' being y; suppose I do this again, with the head being z and the tail w. Then, if I haven't just divided it the same way twice, I'm guaranteed to have a certain kind of overlap between the parts. One option is that x overlaps w in some part u, so that $x = z \frown u$ and $w = u \frown y$; the other option is similar. (Drawing a picture helps.)

axioms as $\overline{\exists} \neq x \frown y$, $\overline{\vee} \neq x \frown y$, and $\overline{\vee} \neq \overline{\exists}$,[27] and similarly for the other primitive expressions of $\mathcal{L}$.[28] This theory is strong enough to prove the Diagonal Lemma (Grzegorczyk, 2005, pp. 228–9).[29]

TC is weak in the sense that it is of the same strength as Robinson arithmetic, usually known as Q.[30] The same goes for $\mathsf{TC}_{\mathcal{L}}$. As it happens, TC is in some ways *too* weak: One cannot even define a pairing function in TC. But Albert Visser (2009), to whom that observation is due, has shown how to augment TC in such a way that a reasonable theory of sequences can be developed, and the augmented theory is still of the same strength as Q.

Note also that, if we have concatenation, then we do not really need corners.[31] It would suffice to have terms denoting each of the primitive expressions of the language. Names of compound expressions can then be constructed using concatenation: We can e.g. replace $\ulcorner \exists x \urcorner$ with $\overline{\exists} \frown \overline{x}$. Since concatenation is associative, strings of more than three symbols will have many such names; we can specify that 'the' quote name of a string associates to the left. For the rest of this section, I'll use corners to abbreviate quote names—or, better, 'canonical' names—of this new kind. The symbol $\mathsf{q}(x)$ can now be thought of as mapping an expression to its canonical name.

More interestingly, once we have concatenation, we do not really need

[27] Here, I'm using e.g. $\overline{\exists}$ as a primitive name of the existential quantifier.

[28] This adds no real strength, since we can code symbols other than the two primitives in terms of them (Grzegorczyk, 2005, §4).

[29] The definition of substitution in TC is somewhat indirect. Rather than giving an explicit definition of the form:

$$\mathsf{subst}(t, u) \overset{df}{=} \phi(t, u)$$

what we actually define is the *graph* of subst:

$$\mathsf{subst}(t, u) = v \;\equiv\; \Phi(t, u, v)$$

We can then treat $\mathsf{subst}(t, u)$ as, in effect, meaning: the v such that $\Phi(t, u, v)$, and eliminate the description as Russell taught. This has the effect, as we'll discuss in Section 2.2, of allowing us only to prove the 'weak' version of the Diagonal Lemma, not the 'strong' version.

[30] For more of the history, and additional references, see Visser (2009, §1.4). The axioms of Q are: $0 \neq Sx$, $Sx = Sy \to x = y$, $x + 0 = x$, $x + Sy = S(x + y)$, $x \times Sy = (x \times y) + x$, and $x \neq 0 \to \exists y(x = Sy)$. This is a very weak theory. It does not, for example, prove the associativity or commutativity of addition or multiplication. It does not even prove $x \neq Sx$.

[31] For a detailed development of this sort of idea, see Grzegorczyk (2005, §6) and Halbach and Leigh (2022, §8.1).

substitution (even if we can define it). This observation seems to be due to Alfred Tarski. In his famous paper "The Concept of Truth in Formalized Languages", Tarski develops his definition of truth within what is, in effect, a theory of syntax. More precisely, Tarski works in a meta-theory that extends the language for which the definition is being given—the 'object-language'—by adding both a theory of the syntax of that language and—for the purpose of defining truth—certain higher-order resources not present in the original language (Tarski, 1956, §§2–3). Tarski does not formalize the meta-theory, but he does state five axioms for the syntactic part of the theory with sufficient precision that formalization is entirely routine (and would have been so when Tarski was writing).

In Tarski's presentation, the only primitive operation is that of concatenation. Tarski also helps himself to terms for each of the primitive expressions in the calculus of classes, much as we have just done; that gives him names for all strings formed from those primitives. When, late in the paper, Tarski proves what we now know as Tarski's Theorem, he has to construct a self-referential sentence.[32] Rather than use substitution to do so, he simply uses concatenation (Tarski, 1956, p. 250).[33] Consider again a formula $A(x)$. The formula

$$\exists x(x = t \wedge A(x))$$

is logically equivalent to $A(t)$, i.e., to $\mathsf{subst}(\ulcorner A(x)\urcorner, \ulcorner t\urcorner)$.[34] So, rather than define diagonalization in terms of substitution, we can simply define it as follows:[35]

$$\mathsf{diag}(z) \overset{df}{=} \ulcorner \exists x(x =\urcorner \frown \mathsf{q}(z) \frown \ulcorner\wedge\urcorner \frown z \frown \ulcorner)\urcorner$$

[32] Tarski (1956, p. 247, n. 1) acknoweldges his debt to Gödel here, noting that this result was added to the paper only after he became aware of Gödel's work.

[33] This construction is more explicit in Tarski et al. (1953, p. 47), where it is used to prove a generalization of the First Incompleteness Theorem. Grzegorczyk (2005, pp. 225–8) gives a similar construction. He first observes that every formula with one free variable is equivalent to one in which the only free variable occurs only free, and only once. (The formula we want is $\exists x(x = v \wedge A(x))$, where v is any variable that does not occur in $A(x)$.) He then notes that defining substitution for such formulas is easy. But Tarski's method is even easier.

[34] Here the convention of omitting quotes around formulas might be confusing. I am here *using* "$\mathsf{subst}(\ulcorner A(x)\urcorner, \ulcorner t\urcorner)$", not mentioning it: $A(t)$ is equivalent to the thing "$\mathsf{subst}(\ulcorner A(x)\urcorner, \ulcorner t\urcorner)$" names, not (of course) to the term itself.

[35] There are some use–mention subtleties here, which I'll confine to the notes. First, let's be clear that $\ulcorner\mathsf{diag}(z)\urcorner$ is itself a term, and it is a name of a formula, namely, the one named by the term on the right-hand side of the definition I'm about to state.

Or less precisely but a bit more clearly:

$$\mathsf{diag}(\ulcorner A(x) \urcorner) \overset{\mathit{df}}{=} \ulcorner \exists x(x = \ulcorner A(x) \urcorner \wedge A(x)) \urcorner$$

To prove the Diagonal Lemma now, we need to start with a formula $A(y)$ containing just y free and consider:

(15) $\mathsf{diag}(\ulcorner \exists y(y = \mathsf{diag}(x) \wedge A(y)) \urcorner)$

Applying the mentioned operation then yields:[36]

(16) $\exists x(x = \ulcorner \exists y(y = \mathsf{diag}(x) \wedge A(y)) \urcorner \wedge \exists y(y = \mathsf{diag}(x) \wedge A(y)))$

which is logically equivalent to

(17) $\exists y(y = \mathsf{diag}(\ulcorner \exists y(y = \mathsf{diag}(x) \wedge A(y)) \urcorner) \wedge A(y))$

and so to[37]

(18) $A(\mathsf{diag}(\ulcorner \exists y(y = \mathsf{diag}(x) \wedge A(y)) \urcorner))$

And now, once again, the argument to $A(\cdot)$ in (18) is just (15), which names (16), which is logically equivalent to (18).[38]

There is a difference, however, in what this construction delivers: (15) does not actually denote (18). Rather, (15) denotes (16). So (18) does *not* 'say of itself' that it has whatever property $A(x)$ expresses. The 'self-referential sentence' is (16), and it is self-referential in two senses:

[36] That is, the term displayed at (15) is a name of the sentence displayed at (16). More formally, what we have is:

$$\begin{gathered}\mathsf{diag}(\ulcorner \exists y(y = \mathsf{diag}(x) \wedge A(y)) \urcorner) = \\ \ulcorner \exists x(x = \ulcorner \exists y(y = \mathsf{diag}(x) \wedge A(y)) \urcorner \wedge \exists y(y = \mathsf{diag}(x) \wedge A(y))) \urcorner\end{gathered}$$

(Sorry about that.)

[37] Note how the work previously done by substitution has been off-loaded onto the logic.

[38] Formally, what we have is that, for each formula $A(y)$ containing just y free, there is a sentence G_A that is provably equivalent to $A(\ulcorner G_A \urcorner)$. The sentence in question is (16). That it is equivalent to

$$A(\ulcorner \exists x(x = \ulcorner \exists y(y = \mathsf{diag}(x) \wedge A(y)) \urcorner \wedge \exists y(y = \mathsf{diag}(x) \wedge A(y))) \urcorner)$$

sorry again—then follows from (i) the fact that (16) is logically equivalent to (18) and (ii) the identity mentioned in note 36, which is a provable truth of syntax (not logic).

1. The formula (18) most certainly does say of (16) that it has whatever property $A(x)$ expresses. But (16) is logically equivalent to (18). So that gives us a weak sense in which (16) 'says of itself' that it has that property: It's logically equivalent to a sentence that says that (16) has whatever property $A(x)$ expresses.

2. What (16) says is that *there is* a sentence which is the diagonalization of a certain expression, and that *that* sentence has whatever property $A(x)$ expresses; as it happens, the sentence in question is (16) itself. In effect, then, (16) refers to itself 'by description' rather than in the more direct way that, say, (14) refers to itself.[39]

As Kripke (1975, p. 692) notes, in that respect, (16) is similar to so-called 'empirical' liars, like

> The only sentence in a displayed quotation on page 16 of 'The Liar Paradox and Metamathematics' is false.

But, in the case of (16), it's a *syntactic* (and so, presumably, necessary) truth that the sentence it describes is (16) itself.

1.4 Quotation

So the construction of self-referential sentences can proceed entirely in terms of concatenation: Nothing so complicated as substitution is required. We do, however, also need access to a function like $\mathsf{q}(x)$. Note that this is different from just being able to form quote names (or canonical names) on a case-by-case basis. What we need, rather, is to be able to describe that operation quite generally: to be able to refer to the function that maps a given expression to its quote name. Such a function is used in every one of the constructions we have so far discussed.

One might think this was quite trivial.[40] If we had quotation marks, for example, couldn't we just define $\mathsf{q}(x)$ as: $\ulcorner\text{“}\urcorner \frown x \frown \ulcorner\text{”}\urcorner$? So can't we just add quotation marks, or something of the same sort?[41] Unfortunately, no, not if we want (a) to be able to form a quote name of every

[39] For some discussion of this kind of self-reference, see Picollo (2018).

[40] Another reason to think it's trivial is that the axioms governing this operation are so easy to state. See note 16.

[41] Similarly, it is easy to formulate a single axiom that governs the semantics of quotation marks:

> For every string x, $\ulcorner\text{“}\urcorner \frown x \frown \ulcorner\text{”}\urcorner$ denotes x.

This does not show that quotation is compositional, since the meaning of a quote name does not depend upon the *meaning* of its parts. (This follows trivially from the fact that string and (b) to have the resulting expressions be unambiguous, which is absolutely essential. Suppose, for example, that we wanted to form a quote name of this expression:

$$\overline{x} \frown \text{“}y\text{”} \frown \text{“}z\text{”}$$

We’d get:

$$\text{“}\overline{x} \frown \text{“}y\text{”} \frown \text{“}z\text{””}$$

But that could instead be parsed as:

$$[\text{“}\overline{x} \frown \text{“}y\text{”}] \frown [\text{“}z\text{””}]$$

where the first expression is a name of: $\overline{x} \frown$ “y, and the second is a name of: z”. George Boolos (1998, p. 395) credits this observation to a student of his, Michael Ernst. As Boolos notes, adding more types of quotation marks seems unlikely to solve the problem, at least so long as these are finite in number.

There is a solution: Boolos shows how we can construct infinitely many types of quotation marks from the two symbols $'$ and $\circ$ (much as infinitely many variables can be constructed from the two symbols x and $'$). One can then make sure, when quoting a string, always to use quotation marks that are ‘longer’ than any contained in that string. But this greatly complicates the defintion of $\mathsf{q}(x)$. Grzegorczyk (2005) addresses a form of this problem, however, and shows that it is solvable.[42] The reason, in brief, is that his theory TC is strong enough to represent all ‘discernible’ functions, just as Q is strong enough to represent all recursive functions (of which the discernible functions are the syntactic analogue), and $\mathsf{q}(x)$ is certainly discernible.

There is, however, a simpler solution, if we are willing to allow only *well-formed* expressions to be quoted.[43] Under this restriction, we can just use quotation marks. I’ll use guillemets (French quotes) for clarity. Thus, “«x»” is a name of the variable “x”. The crucial point is that, if what’s between the guillemets has to be well-formed, then it will

it is possible to quote strings that have no meaning.) But it does show that Donald Davidson (1984) was wrong to be worried that the semantics of quotation cannot be finitely axiomatized.

[42] Much the same problem arises when one tries to build a theory of sequences using concatenation. If we had commas, then the sequence of two strings could just be the two strings separated by a comma. But what if one of the strings contains a comma? For answers, see Quine (1946) and Visser (2009).

[43] And for present purposes, there is no need whatsoever to quote arbitrary strings.

already have balanced guillemets, and be uniquely readable, and there can be no ambiguity. The exceptions, of course, are the cases where a guillemet itself is quoted: «««» and «»»». We could just ban these and instead use primitive names $\bar{«}$ and $\bar{»}$ for the two quote symbols. But we needn't. These two exceptions are readily identified and can be handled straightforwardly: There is only one way to parse expressions containing them so that those expressions are themselves well-formed, and so that all quoted expressions are also well-formed.[44] The function that maps an expression to its quote name can then be defined explicitly as:

$$\mathsf{q}(x) := (\text{«««»} \frown x) \frown \text{«»»»}$$

It is not the prettiest thing in the world, but it works.

The moral of this section, then, is this. The legitimacy of self-referential sentences is indeed beyond doubt. Their existence follows from very basic facts about syntax. Exactly what kinds of self-referential sentences we can construct will depend upon exactly what resources we allow ourselves. But the worst-case scenario is provable equivalence: Given $A(x)$, elementary facts about how concatenation and quotation work suffice to guarantee the existence of a sentence G_A such that G_A is provably equivalent to $A(\ulcorner G_A \urcorner)$.

2 The Diagonal Lemma in Theories of Arithmetic

2.1 Gödel Numbering and Interpretability

Gödel's original construction of a self-referential sentence is carried out, not in a theory of syntax, but rather in a theory of what is, in effect, higher-order arithmetic. As many readers will already know, he is able to do this because of another technique he invented: what we now call 'Gödel numbering' or 'coding'.[45] There are several different ways one can

[44] The point is that «» is not itself well-formed: We do not have a quote name of the empty string, since the empty string is not well-formed. So «» cannot occur as part of a well-formed expression except in one of the two contexts just mentioned, and «««»» cannot occur at all. (If we wanted a name of the empty string, we could add a primitive term for it, as Visser (2009) does.) So we can, in effect, just treat «««» and «»»» as if they were primitive symbols.

[45] It would be difficult to over-state the importance of Gödel numbering. It is absolutely fundamental to computer science. It is, for example, what allows a machine that operates with 0s and 1s to compute with strings.

think of Gödel numbering. Tarski (1956, p. 184) seems to have thought of it as a way of 'interpreting' syntax in arithmetic.

From one perspective, interpretation, in this sense, is a technique for producing 'relative consistency proofs'.[46] Given some 'base' theory $\mathcal{B}$ and some 'target' theory $\mathcal{T}$—stated in languages $\mathcal{L}_\mathcal{B}$ and $\mathcal{L}_\mathcal{T}$, respectively—one interprets $\mathcal{T}$ in $\mathcal{B}$ by showing how $\mathcal{L}_\mathcal{T}$ can be translated (in a purely formal sense) into $\mathcal{L}_\mathcal{B}$. The translation is compositional, in the sense that the only thing we actually need to do is define the (non-logical) primitive expressions of $\mathcal{L}_\mathcal{T}$ in terms of those of $\mathcal{L}_\mathcal{B}$ and specify a 'domain' for the interpretation in terms of a formula $\delta(x)$ of $\mathcal{L}_\mathcal{B}$.[47,48] This can then be extended to a complete translation of $\mathcal{L}_\mathcal{T}$ into $\mathcal{L}_B$ in the obvious way. For example: $(A \wedge B)^*$ = $(A^*) \wedge (B^*)$, where A^* is the translation of A. Quantifiers are 'relativized' to $\delta(x)$: $\forall x(\phi(x))$ is translated as: $\forall x(\delta(x) \rightarrow \phi^*(x))$; $\exists x(\phi(x))$, as: $\exists x(\delta(x) \wedge \phi^*(x))$. One then completes the interpretation by showing that the translations of the axioms of $\mathcal{T}$ can be proven in $\mathcal{B}$. We also need proofs of $\delta(t^*)$, for each primitive term t of $\mathcal{L}_\mathcal{T}$ (if any), and of the closure condition

$$\forall x_1 \cdots x_n[\delta(x_1) \wedge \cdots \wedge \delta(x_n) \rightarrow \delta(f^*(x_1, \ldots, x_n))]$$

for each primitive function-symbol f, of however many places. We also need (if this isn't already covered) a proof that the domain is non-empty: $\exists x(\delta(x))$.

The crucial fact is then that, because the translation preserves logical form, the translations of the *theorems* (as well as the axioms) of $\mathcal{T}$ can

[46] The first detailed study of interpretation is in Tarski, Mostowski, and Robinson (1953), but the notion is much older. (Their focus is on proofs of undecidability, which implies incompleteness.) Proofs of the consistency of non-Euclidean geometries, given in the 19th century, use this technique: They show how non-Euclidean geometries can be interpreted in Euclidean geometry.

[47] We allow terms and function-symbols to be translated using descriptions, which can then be eliminated as Russell (1905) taught. In that case, we need $\mathcal{B}$ to prove that the descriptions are proper. This technique is essential, for example, for interpreting PA in ZF: There are no terms in the language of set theory other than variables, so 0 cannot really be interpreted as (say) $\emptyset$ but must be interpreted via the description $\iota x \forall y(y \notin x)$. Similarly, $x + y$ has to be interpreted via some formula $\mathsf{Sum}(x, y, z)$. In this case, of course, formulas like $\delta(t^*)$ will be far more complex than this notation makes them appear. To avoid this complication, it is sometimes convenient to work first with purely relational langauges—no terms, no function-symbols—and then to reduce non-relational languages to relational ones. See Visser (1998) for more on this issue.

[48] What I'm about to describe is a one-dimensional relative interpretation without parameters. There are more general notions of interpretation, but we'll not need them here.

also be proven in $\mathcal{B}$. Roughly: To translate a $\mathcal{T}$-proof, one first proves the translations of the needed axioms of $\mathcal{T}$ and then appends a (slightly modified) version of the original $\mathcal{T}$-proof. So, quite generally, if $\mathcal{T} \vdash A$, then $\mathcal{B} \vdash A^*$, where, again, the asterisk means: translation of. It follows that, if $\mathcal{B}$ is consistent, so is $\mathcal{T}$: If $\mathcal{T} \vdash A \wedge \neg A$, then $\mathcal{B} \vdash (A \wedge \neg A)^*$, so $\mathcal{B} \vdash A^* \wedge (\neg A)^*$, so $\mathcal{B} \vdash A^* \wedge \neg(A^*)$, and $\mathcal{B}$ is inconsistent, too.

For our purposes, though, a different way of thinking of interpretation is more useful: Interpreting $\mathcal{T}$ in $\mathcal{B}$ is a way of making the resources of $\mathcal{T}$ available in $\mathcal{B}$. For example, although Peano arithmetic is, in the first instance, a theory about natural numbers, we can code finite sets of numbers bit-wise: S is coded by the number whose binary representation has a 1 in the $n+1^{\text{th}}$ place (counting from the right) just in case $n \in S$. One can then go on to define membership and various set theoretic operations such as union and intersection. All of that then allows us to 'talk', in PA, about finite sets of numbers by instead talking about their codes.[49]

Gödel numbering is a way of doing the same thing for syntax. Given any (countable) language $\mathcal{L}$, we can interpret $\mathsf{TC}_{\mathcal{L}}$ in arithmetic by (i) specifying a translation for each of the primitive names of $\mathsf{TC}_{\mathcal{L}}$ (e.g., $\overline{\exists}$ and $\overline{\vee}$) and (ii) showing how to define concatenation so that the translations of the axioms of $\mathsf{TC}_{\mathcal{L}}$ will be provable in whatever arithmetical theory we are considering. And, as mentioned earlier, we already know that $\mathsf{TC}_{\mathcal{L}}$ is interpretable in Q. So the resources of a weak theory of syntax, for any countable language $\mathcal{L}$, can be made available in any arithmetical theory containing Q. In particular, if $\mathcal{A}$ is the language of arithmetic, then $\mathsf{TC}_{\mathcal{A}}$ is interpretable in Q, which allows Q to 'talk about' its own syntax. Moreover, the proof of the Diagonal Lemma for $\mathsf{TC}_{\mathcal{A}}$ will 'lift' into Q, so we'll get a proof of the Diagonal Lemma (under the translation) for Q.

Gödel numberings are usually presented somewhat differently, in two respects.[50] Gödel's own procedure makes for a good example. First, Gödel (1986, p. 161) specifies an association between *symbols* and *numbers*, not a way of translating names of symbols by names of numbers. So, e.g., the code for $\vee$ is the number 7, in Gödel's treatment. Second, Gödel does

[49] We can do much better: The so-called Ackermann coding allows us to code hereditarily finite sets of numbers; arithmetically definable sets can be coded by the Gödel numbers of formulas that define them; and so forth. The standard reference for such things is Hájek and Pudlák (1993).

[50] Actually, three: No independent (so to speak) syntactic theory is usually mentioned at all.

not just specify a code for each of the primitive symbols but specifies such a code for every string. Given such an association between symbols and numbers, there will then be some number-theoretic function that corresponds to concatenation (and to every other syntactic operation): the image of that function under the coding.[51] Gödel then proceeds to show—after the fact, as it were—that the number-theoretic analogue of concatenation is definable in P (Gödel, 1986, p. 165).

There is nothing wrong with this approach, but it does have the disadvantage that it leaves it somewhat mysterious exactly what restrictions we might want to impose upon the coding, and why. Here's an extreme example. Tarski's Theorem is often stated as: Arithmetical truth is not arithmetically definable. But, actually, that's not true. It *is* possible to define arithmetical truth if one uses the right kind of Gödel numbering. Let $s_0, s_1, \ldots$ be a list of all well-formed expressions of the language of arithmetic. If s_0 is a true sentence of that language, then let its code be 0; otherwise, let it be 1. More generally, if s_i is a true sentence, then let its code be the next even number; otherwise, the next odd number. Then $\exists y(x = SS0 \times y)$—i.e., x is even—defines arithmetical truth.[52]

Is that cheating? In some sense, yes, but in what sense? The statement of Tarski's Theorem in Tarski, Mostowski, and Robinson's *Undecidable Theories* addresses this lacuna. They define a 'diagonal function', much as we have above, and state the theorem as: If a theory $\mathcal{T}$ (which need not be recursively axiomatizable) is consistent, then the set of theorems of $\mathcal{T}$ and the diagonal function are not both definable in $\mathcal{T}$ (Tarski et al., 1953, p. 46).[53] So we can conclude that, under the coding just described, the diagonal function will not be definable. But one might wonder why that should be desirable. Who cares if the diagonal function is definable? Maybe it would be better if it weren't. The best answer to this question, it seems to me, is that we want *concatenation* to be definable, and for its elementary properties to be provable, because we want to be able to 'do syntax' in arithmetic. But that is, essentially, to say that we want to be able to interpret a reasonable theory of syntax. For some

[51] I.e., let $\frown$ be concatentation in its syntactic sense, and let $g(\cdot)$ be the coding function. Then the 'number-theoretic analogue' of concatentation is that function $*$ such that $g(s) * g(t) = g(s \frown t)$, for all strings s and t. This function will usually be partial; we can stipulate a throwaway value for the other cases, if we wish.

[52] For a more sophisticated construction in the same spirit, see Visser (2004, pp. 164–5).

[53] By 'definable', Tarski, Mostowski, and Robinson (1953, p. 45) mean 'representable', in the sense to be defined below.

purposes, of course, we may want to impose stronger conditions, e.g., that the image of the concatenation function should be recursive (see e.g. Tarski et al., 1953, p. 48). But that condition, while not unmotivated—surely concatenation is a computable operation—is still stronger than is needed for Tarski's Theorem. Moreover, it is not easily stated for theories in non-arithmetical languages.[54]

We get the right level of generality, I suggest, if we state Tarski's Theorem in terms of interpretability.[55]

Theorem (Tarski's Theorem)**.** *Let $\mathcal{T}$ be a theory in the language $\mathcal{L}$, and let $\mathcal{I}$ be an interpretation of $\mathsf{TC}_{\mathcal{L}}$ in $\mathcal{T}$. Then, if $\mathcal{T}$ is consistent, there is no formula $\mathsf{Tr}(x)$ of $\mathcal{L}$ such that, for each sentence A of $\mathcal{L}$, $\mathcal{T} \vdash \mathsf{Tr}(\ulcorner A \urcorner^{\mathcal{I}}) \equiv A$, where $\ulcorner A \urcorner^{\mathcal{I}}$ is the $\mathcal{I}$-translation of the canonical name of A in $\mathsf{TC}_{\mathcal{L}}$.*

The proof is a straightforward adaptation of the usual proof, making use of the interpretation of $\mathsf{TC}_{\mathcal{L}}$ (and the availability of the diagonal lemma in $\mathsf{TC}_{\mathcal{L}}$) to produce a sentence L such that $L \equiv \neg\mathsf{Tr}(\ulcorner L \urcorner^{\mathcal{I}})$ is a theorem of $\mathcal{T}$.

Another advantage of thinking of arithmetization in terms of interpretability is that it extends smoothly to non-arithmetical theories, such as various forms of set theory. Of course, one can 'do Gödel numbering' in ZF by first developing arithmetic in ZF and then using that to mimic one's favorite arithmetical coding. But that kind of indirection is obviously inessential. Alternatively, then, one can code symbols as sets directly (e.g., the code of $\vee$ might be the empty set). But now, what restrictions do we want to put on how this coding is done? The answer, I suggest again, is most naturally given in terms of the interpretability of a certain elementary theory of syntax, such as TC.

[54] Indeed, there is an analogous problem concerning the 'coding' of arithmetic on Turing machines: If we represent numbers using a non-standard coding, we can arrange for e.g. the halting function to be computable. See Rescorla (2007) and references therein for discussion.

[55] In fact, this can be strengthened. As Grabmayr (2021) shows, even the requirement that concatenation be provably total is stronger than it needs to be. But that will be required if we're to interpret TC, since it contains a function-symbol for concatenation. As mentioned earlier, however, whereas TC is a kind of analogue of Q, there is another theory in the same language that would be an analogue of R, and that will be enough. Moreover, it is almost certainly enough to be able to interpret a *relational* version of TC that does not assume existence and uniqueness as general principles but only tells us that, in each concrete case, the concatenation of two expressions exists and is unique. See Švejdar (2007) and Heck (2014) for discussion of the arithmetical case.

$($	$)$	$\exists$	$\forall$	$\vee$	$\wedge$	$\rightarrow$	$\neg$	x	$'$	0	S	$+$	$\times$	$=$
1	2	3	4	5	6	7	8	9	a	b	c	d	e	f

Table 1: The Basic Correspondence

2.2 Gödel Numbering, the Usual Way

As said, Gödel numbering is not usually done the way just described. It is, therefore, worth being acquainted with the usual sort of treatment. The work we have done so far will allow us to isolate one of the otherwise puzzling elements that comprises it.

We begin, of course, by establishing some one–one correspondence between strings and numbers. Here is one simple way of doing that.[56] Start with the correspondence between primitive symbols and hexadecimal (base-16) digits in table 1. We extend the correspondence to one between strings of symbols of the language of arithmetic and hexadecimal numerals by treating these symbols as if they just were the corresponding hexadecimal digits. Thus, for example, we read the string '$) + 0\exists\vee$' as if it just were the hexadecimal numeral '$2db35$'. This induces a correlation between strings of symbols and natural numbers. The one-element string '0' is thus correlated with the number b_{16}, or 11; the string '$) + 0\exists\vee$' is correlated with the number $2db35_{16}$, or $187,189$. It's then easy enough to see which number-theoretic function is the image of concatenation under this mapping:[57]

$$\mathsf{Concat}(n, m) := n \times 16^{\mathsf{log}_{16}(m)+1} + m$$

It's less obvious how this function might actually be defined in the language of arithmetic.

Some authors, including Gödel (1986) and Boolos (1993), actually produce explicit definitions of the concatenation function (or something close enough), and ultimately of the diagonalization function. But it is more common nowadays to proceed differently.[58] It's a quite general fact about Q that every recursive function is 'representable' in Q. Here's what that means.

[56] Something like this construction is also presented by Smullyan (1982, §4).

[57] Here, $\mathsf{log}_{16}(m)$—the (truncated) base 16 logarithm—will be one less than the length of the hexadecimal numeral for m. So $\mathsf{Concat}(n, m)$ is the number whose hexadecimal representation is given by that for n followed by that for m.

[58] This approach was popularized by Tarski et al. (1953), which is in many ways the first really modern treatment of the incompleteness theorem—though it draws upon the exposition in Mostowski (1952).

Definition. Let $\mathcal{T}$ be a theory in the language of arithmetic. Then the function $f(x)$ is *represented* in $\mathcal{T}$ by the formula $F(x,y)$ just in case, whenever $f(n)=m$:

1. $\mathcal{T} \vdash F(\overline{n}, \overline{m})$
2. $\mathcal{T} \vdash F(\overline{n}, x) \rightarrow x = \overline{m}$

Or, equivalently:

3. $\mathcal{T} \vdash F(\overline{n}, x) \equiv x = \overline{m}$

Similarly for functions of more than one argument.

Here, $\overline{n}$ is the numeral for n, i.e., $S \ldots S0$, where there are n Ss. Note that what's required is that $\mathcal{T}$ should prove that $f(\overline{n})$ has a value *in each case*, and that it should prove *in each case* that the value is unique. It is *not* required that $\mathcal{T}$ should prove existence or uniqueness as a general principle, i.e., that it should prove $\forall x \exists y F(x,y)$ or $\forall x \forall y \forall z(F(x,y) \wedge F(x,z) \rightarrow y = z)$. This is important. Weak theories like Q (let alone R) will not be able to prove such generalizations.[59]

Note that a one-place function is 'represented' by a two-place formula, i.e, by a relation. In effect, what we are defining isn't $f(x)$ but $y = f(x)$; that relation is sometimes called the 'graph' of $f(x)$. The reason we must do this is that the language of arithmetic is term-poor: There are many primitive recursive functions, such as 2^x, that cannot be defined by any term, however complex, formed from just 0, S, $+$, and $\times$.[60] But there is a formula $\mathsf{Exp}(x,y)$ that *represents* 2^x, already in Q, in the sense that Q proves each instance of the conditions mentioned. The key result, as already said, is then that every recursive function is representable in Q. Since the image of concatenation is recursive,[61] we can thus conclude that it is representable in Q; similarly for substitution and quotation.[62] Indeed, since the diagonalization function (defined in any of the ways we have discussed) is recursive, it too will be representable in Q.

[59] In fact, the definition allows for the possibility that $f(x)$ should be partial. But even when it is not, we do not require $\mathcal{T}$ to prove existence and uniqueness.

[60] Any such term is equivalent to a polynomial, and every polynomial is eventually dominated by 2^x.

[61] Quite generally, if the coding function is recursive and has a recursive inverse, then the image of every 'discernible' syntactic function will be recursive. This is easy to see using Church's Thesis.

[62] In this case, the quotation function can be taken just to be the numeral function: the function that takes us from a number to the code of the numeral that denotes it. That numeral is the canonical name of the number in question.

Typically, then, the diagonal function will be formally encoded as a *relation*, not as a function. This is one of the technical subtleties that can make the Diagonal Lemma harder to understand than it needs to be. The way I remember the proof of the Diagonal Lemma is as involving the formula $A(\mathsf{diag}(x))$, which we then diagonalize to get $A(\mathsf{diag}(\ulcorner A(\mathsf{diag}(x))\urcorner))$. That then turns out to be what is named by the embedded argument $\mathsf{diag}(\ulcorner A(\mathsf{diag}(x))\urcorner)$. But there isn't really a term $\mathsf{diag}(x)$ in the language of arithmetic. Rather, there is a formula $\mathsf{Diag}(x, y)$ that represents diagonalization. So $A(\mathsf{diag}(x))$ becomes:

(19) $\exists y(\mathsf{Diag}(x, y) \wedge A(y))$

and its diagonalization (if defined in terms of substitution) is:

(20) $\exists y(\mathsf{Diag}(\ulcorner \exists y(\mathsf{Diag}(x, y) \wedge A(y))\urcorner, y) \wedge A(y))$

We then reason as follows. Let d be the Gödel number of (20). Assuming that $\mathsf{Diag}(x, y)$ represents diagonalization in $\mathcal{T}$, we thus have:

(21) $\mathcal{T} \vdash \mathsf{Diag}(\ulcorner \exists y(\mathsf{Diag}(x, y) \wedge A(y))\urcorner, y) \equiv y = \overline{d}$

So (20) is $\mathcal{T}$-provably equivalent to $\exists y(y = \overline{d} \wedge A(y))$[63] and so to $A(\overline{d})$. That is, roughly:

$$\mathcal{T} \vdash \text{(20)} \equiv A(\ulcorner \text{(20)} \urcorner)$$

or, more precisely:

$$\begin{gathered}\mathcal{T} \vdash \exists y(\mathsf{Diag}(\ulcorner \exists y(\mathsf{Diag}(x, y) \wedge A(y))\urcorner, y) \wedge A(y)) \equiv \\ A(\ulcorner \exists y(\mathsf{Diag}(\ulcorner \exists y(\mathsf{Diag}(x, y) \wedge A(y))\urcorner, y) \wedge A(y))\urcorner)\end{gathered}$$

which gives us the Diagonal Lemma in the form that asserts provable equivalence: There is a sentence G_A such that $\mathcal{T} \vdash G_A \equiv A(\ulcorner G_A \urcorner)$. The wanted sentence G_A is just (20).

Note, as earlier, that $A(\overline{d})$ does not refer to itself. It refers, rather, to (20), which refers to itself only 'by description': (20) says that *there is* a sentence meeting a certain condition, and that *that* sentence has whatever property $A(x)$ expresses. That sentence just so happens to be (20).

One might wonder, then, whether we can do better: Whether we can produce a proof of the Diagonal Lemma that doesn't involve this kind of indirection. As we'll see, there are a few ways to do that, some of them better than others.

[63] Because the left-hand side of (21) is the first conjunct of (20). So we can replace that conjunct with $y = \overline{d}$.

2.3 The Strong Diagonal Lemma

One way to proceed is to consider a richer language.[64] The theory known as Primitive Recursive Arithmetic is formulated in a language that has function symbols for every primitive recursive function (and has, among its axioms, the equations that define that function). Typical Gödel numberings, such as the one mentioned above, make (the images of) concatenation, substitution, and diagonalization not just recursive but primitive recursive. So, in PRA, there will be function symbols for all of these functions. In particular, there will be a function symbol $\mathsf{diag}(x)$ that defines diagonalization, and the axioms of PRA are strong enough for us to prove, in each particular case, what the diagonalization of any given formula is. In particular, $\mathsf{PRA} \vdash \mathsf{diag}(\ulcorner A(x) \urcorner) = \ulcorner A(\ulcorner A(x) \urcorner) \urcorner$,[65] for each formula $A(x)$.

With that in hand, we can then prove the Diagonal Lemma more straightforwardly. Now there really is a formula $A(\mathsf{diag}(x))$, and we can consider the term:

(22) $\mathsf{diag}(\ulcorner A(\mathsf{diag}(x)) \urcorner)$

The diagonalization of $A(\mathsf{diag}(x))$ is, familiarly:

(23) $A(\mathsf{diag}(\ulcorner A(\mathsf{diag}(x)) \urcorner))$

So PRA proves:

$$\mathsf{diag}(\ulcorner A(\mathsf{diag}(x) \urcorner) = \ulcorner A(\mathsf{diag}(\ulcorner A(\mathsf{diag}(x)) \urcorner)) \urcorner$$

or, roughly:

$$(22) = \ulcorner A(\ulcorner (22) \urcorner) \urcorner$$

So now (23) really does refer to itself: The embedded argument refers to (23).

So, in PRA, we have what is sometimes known as the 'Strong' Diagonal Lemma (though it might better be called the 'term' form of the Lemma).

[64] This way of proving the Diagonal Lemma first appears in print, so far as I can tell, in Jeroslow (1973). Halbach and Visser (2014a, p. 684) suggest, though, that Kreisel was aware of it by 1953.

[65] Note that this is all we really need. So the theory that Halbach and Visser (2014a, pp. 674–5) call Basic will suffice. Basic extends R by adding function symbols for all primitive recursive functions and, as new axioms, all true identities of the form: $\overline{n} = t$, where t is a closed term. As they note, Basic is interpretable in R. (We can, presumably, also work in a relational version of Basic that does not assume existence or uniqueness as general principles, but only that values exist and are unique in each case.)

Lemma (Strong Diagonal Lemma)**.** *For each formula $A(x)$, there is a term t_A such that* $\mathsf{PRA} \vdash t_A = \ulcorner A(t_A) \urcorner$.

The provable equivalence, or Weak (or 'sentence'), form then follows, as we saw earlier, by Leibniz's Law.

Boolos (1993, pp. 24ff) suggests a way of helping ourselves to terms like $\mathsf{diag}(x)$ even when they aren't actually available. The idea is to treat such terms as abbreviations, to be eliminated contextually via Russell's theory of descriptions. So $A(\mathsf{diag}(x))$ abbreviates: $\exists y(\mathsf{Diag}(x,y) \wedge A(y))$.[66] Boolos imposes the condition on the use of such 'pseduo-terms' that existence and uniqueness be provable for them. That guarantees that all 'disabbreviations' of these terms will be provably equivalent, so we don't have to worry about scope issues. And, in the context of the theory in which Boolos is working, namely PA, this restriction is harmless: Existence and uniqueness will be provable for the various syntactic operations we need.[67] In the context of weaker theories, however, such as Q, existence and uniqueness will not be provable. So long as we are only interested in formulas where the argument-places of the term are filled by numerals, though—or by some other closed term—the two conditions that define representability imply that we do have existence and uniqueness for each particular case. So, once again, we need not worry about ambiguity, so long as we stick to closed terms. Observing this restriction, then, we can reason with pseudo-terms as if they were real terms, and unabbreviate only when we are done.

2.4 Some Lessons Worth Learning

Which form of the Diagonal Lemma we have available can make as big a difference as it possibly could: Whether a given theory is consistent can depend upon it. It requires some care to state this fact precisely. Write: $\ulcorner A \urcorner^g$ to mean: the numeral for the Gödel number of A, according to some

[66] Or, in the first instance, we have $\mathsf{diag}(x) = \iota y \mathsf{Diag}(x,y)$, and then we eliminate the description operator. We could add a clause expressing uniqueness to what's in the text, but there's no particular need to do so.

[67] Indeed, if existence and uniqueness are provable, then the theory that results from adding the new axiom $f(x) = y \equiv F(x,y)$ will be a conservative extension of the original theory (exercise!), and we can actually work in that theory. The disabbreviaton of anything we prove in the extended theory will then be provable in the original theory.

particular Gödel numbering g. Then the theory whose axioms are:

$$\begin{aligned} \mathsf{Tr}(\ulcorner \neg A \urcorner^g) &\equiv \neg \mathsf{Tr}(\ulcorner A \urcorner^g) \\ \mathsf{Tr}(t) &\equiv \mathsf{Tr}(\ulcorner \mathsf{Tr}(t) \urcorner^g) \end{aligned}$$

where t is a closed term, is inconsistent if the Gödel numbering is one for which the Strong Diagonal Lemma holds, but it can be consistent if it is one for which only the Weak Diagonal Lemma holds (Heck, 2007, §3.1; Grabmayr and Visser, 2021, §9).[68] In particular, the mentioned theory is consistent if we use the Gödel numbering described above and, indeed, most of the Gödel numberings typically considered in the literature. In so far as the mentioned theory is *intuitively* inconsistent—there's a straightforward informal argument for its inconsistency (Heck, 2007, pp. 12–3)—that might be seen as a reason to prefer formal frameworks in which the Strong Diagonal Lemma is available.

Another advantage of the Strong Diagonal Lemma is that it is relatively independent of the background logic. Consider, for example, Saul Kripke's (1975) theory of truth.[69] The logic of the (best-known) theory Kripke presents is strong three-valued (Kleene) logic. So a biconditional $A \equiv B$ is true just in case A and B are both true or both false, and undefined otherwise. It follows that we do *not* have the usual form of the Diagonal Lemma in Kripke's theory. If we did, then there would be a formula Λ such that $\Lambda \equiv \neg\mathsf{Tr}(\ulcorner \Lambda \urcorner)$. But such a Λ would be paradoxical and so neither true nor false. So the biconditional cannot be true. But there is no obstacle to the Strong Diagonal Lemma, and if Kripke's theory is formulated with the base theory being PRA, then we will indeed have the Strong Diagonal Lemma. Leibniz's Law, in this setting, will take the form of an inference rule:

$$t = u, A(t) \vdash A(u)$$

So we'll end up with a term λ such that $\lambda = \ulcorner \neg\mathsf{Tr}(\lambda) \urcorner$, and Leibniz's Law will imply that $\mathsf{Tr}(\lambda)$ is inter-derivable with $\mathsf{Tr}(\ulcorner \neg\mathsf{Tr}(\lambda) \urcorner)$, and similarly for $\neg\mathsf{Tr}(\lambda)$ and $\neg\mathsf{Tr}(\ulcorner \neg\mathsf{Tr}(\lambda) \urcorner)$.[70]

That is not to say that Kripke's theory has to be formulated in an expressive language like that of PRA. If our base theory is PA, say, and if

[68] It's essential, of course, that A, in the first scheme, be allowed to contain the truth-predicate.

[69] [[XREF to that chapter]]

[70] As I'll emphasize below, it's important to note that, while $\lambda = \ulcorner \neg\mathsf{Tr}(\lambda) \urcorner$, λ is *not* the same term as $\ulcorner \neg\mathsf{Tr}(\lambda) \urcorner$. The latter symbol abbreviates the numeral for the Gödel number of $\neg\mathsf{Tr}(\lambda)$, whereas λ itself is the term $\mathsf{diag}(\ulcorner \neg\mathsf{Tr}(\mathsf{diag}(x)) \urcorner)$.

we use an ordinary sort of Gödel numbering, then we will still have the Diagonal Lemma formulated in terms of inter-derivability: There will be, for each formula $A(x)$, a formula G_A such that G_A is inter-derviable with $A(\ulcorner G_A \urcorner)$.[71] We see, though, how sentential forms of the Diagonal Lemma can be very sensitive to the background logic.

This brings us to an important and often over-looked point. As mentioned earlier, Kripke (1975, p. 692) claims that "self-referential sentences... are as incontestably legitimate as arithmetic itself". That is correct if what Kripke has in mind is the *Strong* Diagonal Lemma (or something similar). Only quite elementary arithmetical (or syntactic) resources are needed to prove it. Since everything involved there is primitive recursive, even finitists will regard it as unproblematic. The same cannot be said of the Weak Diagonal Lemma, even in its inter-derivability form. That does *not* follow just from arithmetic unless the formula involved is itself arithmetical. The proof of the equivalence, whatever form it takes, will involve an appeal to instances of Leibniz's Law for the formula $A(x)$. If that formula is not arithmetical—if, in particular, it is $\neg\mathsf{Tr}(x)$—then arithmetic itself will have nothing to say about it. In that sense, then, arithmetic alone does *not* guarantee the existence of a formula Λ such that Λ is inter-derivable with $\neg\mathsf{Tr}(\ulcorner \Lambda \urcorner)$. The proofs of these claims involve the use, not just the mention, of non-arithmetical resources.

I emphasize this point because it has become common in the literature to begin with the assumption that some sentence L is inter-derivable with some other sentence $A(\ulcorner L \urcorner)$ and then to focus attention on the question what *other* resources are needed to derive a paradox.[72] It's effectively assumed that only those other resources can be questioned, because there is no questioning self-reference. But that is a mistake. There is, to be sure, a sense in which there should be no questioning self-reference. Very basic syntactic and arithmetic facts imply that there are sentences that refer to themselves, whether in the sense that there is a term t_A such that $t_A = \ulcorner A(t_A) \urcorner$ or in the less direct sense of (21). Inter-dervability requires more: The application of Leibniz's Law to $A(x)$. Someone who wanted to question that move would not be questioning elementary arithmetic but the same sort of logical principle that non-

[71] This is because we will still have (21), above; that will imply that (20) is inter-derviable with $A(\overline{d})$.

[72] For example, Murzi and Rossi (2017, p. S825) proceed this way—though one could cite many other papers on the semantic paradoxes. See page 37 for more on why this matters.

classical ‘solutions’ of the semantic paradoxes usually question.[73]

3 Generalizations of the Diagonal Lemma

There are two important generalizations of the Diagonal Lemma. For simplicity, I’ll restrict attention to sentence-based, provable equivalence forms, and I’ll assume that we are working in classical logic.

In the form in which we have been discussing it, which is the form in which it is usually stated, the Diagonal Lemma applies to formulas containing exactly one free variable. In fact, however, that restriction is inessential. Exactly the same construction can be used to prove this generalization:

Lemma. *Let $A(x, y_0, y_1, \dots)$ be a formula containing at least x free. Then, so long as Σ contains R, there is a formula $G_A(y_0, y_1, \dots)$ such that Σ proves the univeral closure of*

$$G_A(y_0, y_1, \dots) \equiv A(\ulcorner G_A(y_0, y_1, \dots)\urcorner, y_0, y_1, \dots)$$

This form of the Diagonal Lemma can be used to formalize Yablo’s Paradox (Yablo, 1993).[74] Informally, the paradox concerns an infinite sequence of sentences $Y_0, Y_1, \dots$, where each of the Y_i says that none of the later sentences in the sequence is true. Let $\mathsf{Sat}(x, y)$ be a simple satisfaction predicate that says, roughly: The one-place formula coded by x is true of y; so, ideally, we would want:[75]

$$\mathsf{Sat}(\ulcorner A(z)\urcorner, y) \equiv A(y)$$

Now consider:

$$\forall y(z < y \rightarrow \neg\mathsf{Sat}(x, y))$$

[73] This type of doubt is usually associated with intentionalist treatments of the liar (Skyrms, 1984) or those that emphasize the role of sentence tokens (Gaifman, 1992; Goldstein, 1992). But also one can think of contextualists (Parsons, 1981; Burge, 1984; Simmons, 1993; Glanzberg, 2001) as questioning this move: Sentence identity does not guarantee identity of truth-value, because it does not guarantee identity of proposition expressed. (Indeed, I’m inclined to think that intentionalist and token-based views collapse, under pressure, into contextualism. See Christman (2023) for discussion.)

[74] Visser (2004, pp. 166–8) presents a closely related construction. This way of formalizing the paradox seems first to have been presented by Priest (1997) and then independently rediscovered by many others, including Ketland (2005) and myself. There are other ways of proceeding, as well (Halbach and Zhang, 2017).

[75] This general principle, with y a variable, is essential to the derivation of a contradiction, as is the generalization (24). See Ketland (2005) and Cook (2014, pp. 25ff) for the reasons.

Use the form of the Diagonal Lemma just mentioned to obtain a formula $Y(z)$ such that:

(24) $Y(z) \equiv \forall y(z < y \to \neg\mathsf{Sat}(\ulcorner Y(z) \urcorner, y))$

Since $\mathsf{Sat}(\ulcorner Y(z) \urcorner, y) \equiv Y(y)$, we have:

$$Y(z) \equiv \forall y(z < y \to \neg Y(y))$$

So $Y(\overline{n})$ is equivalent to $\forall y(\overline{n} < y \to \neg Y(y))$, which says that all of these sentences:

$$\begin{gathered}\forall y(\overline{n+1} < y \to \neg Y(y)) \\ \forall y(\overline{n+2} < y \to \neg Y(y)) \\ \vdots\end{gathered}$$

are false. The usual reasoning then leads to a contradiction. Suppose $Y(\overline{n})$ is true. Then all of the listed sentences are false. But then they are all true! In particular, $\forall y(\overline{n+1} < y \to \neg Y(y))$ is true: It says that all of the sentences on the list except the first are false, and they are. So no $Y(\overline{n})$ can be true. So all the $Y(\overline{n})$ must be false. But then, by the same reasoning, every $Y(\overline{n})$ is true: For $Y(\overline{n})$ says that all the listed sentences are false, and they are. Contradiction.

A second generalization allows for diagonalization on multiple sentences simultaneously.

Lemma. *Let $A_1(x_1, x_2, \ldots, x_n)$, $A_2(x_1, x_2, \ldots, x_n)$, …, $A_n(x_1, x_2, \ldots, x_n)$ be formulas in which exactly the variables shown are free.*[76] *Then there are sentences G_1, G_2, …, G_n such that:*

$$\begin{aligned} G_1 &\equiv A_1(\ulcorner G_1 \urcorner, \ulcorner G_2 \urcorner, \ldots, \ulcorner G_n \urcorner) \\ G_2 &\equiv A_2(\ulcorner G_1 \urcorner, \ulcorner G_2 \urcorner, \ldots, \ulcorner G_n \urcorner) \\ &\vdots \\ G_n &\equiv A_n(\ulcorner G_1 \urcorner, \ulcorner G_2 \urcorner, \ldots, \ulcorner G_n \urcorner) \end{aligned}$$

The proof is straightforward, but not obvious, and not often presented, so I'll give it. Note first, though, that the requirement that all the displayed variables be free can be relaxed. If x_2 isn't free in A_1, say, then we can just replace A_1 with the logically equivalent $A_1 \wedge x_2 = x_2$. But the proof is simpler with the requirement imposed.

[76] One can also allow extra free variables here.

Proof. We'll just do the case of two formulas, and we'll assume, for ease of exposition, that we're working in a rich language, like that of PRA.[77] Let the 1-diagonalization of $A_1(x, y)$ and $A_2(x, y)$ be:

$$A_1(\ulcorner A_1(x, y)\urcorner, \ulcorner A_2(x, y)\urcorner)$$

and let the 2-diagonalization be:

$$A_2(\ulcorner A_1(x, y)\urcorner, \ulcorner A_2(x, y)\urcorner)$$

These are recursive operations, so there are terms diag_1 and diag_2 that represent them. Now consider these formulas:

$$\begin{aligned} A_1^*(x, y) &:= A_1(\mathsf{diag}_1(x, y), \mathsf{diag}_2(x, y)) \\ A_2^*(x, y) &:= A_2(\mathsf{diag}_1(x, y), \mathsf{diag}_2(x, y)) \end{aligned}$$

The wanted sentences are then:

$$\begin{aligned} G_1 &:= A_1(\mathsf{diag}_1(\ulcorner A_1^*\urcorner, \ulcorner A_2^*\urcorner), \mathsf{diag}_2(\ulcorner A_1^*\urcorner, \ulcorner A_2^*\urcorner)) \\ G_2 &:= A_2(\mathsf{diag}_1(\ulcorner A_1^*\urcorner, \ulcorner A_2^*\urcorner), \mathsf{diag}_2(\ulcorner A_1^*\urcorner, \ulcorner A_2^*\urcorner)) \end{aligned}$$

Observe that $\mathsf{diag}_1(\ulcorner A_1^*\urcorner, \ulcorner A_2^*\urcorner)$ is just G_1, and $\mathsf{diag}_2(\ulcorner A_1^*\urcorner, \ulcorner A_2^*\urcorner)$ is just G_2. So we have:

$$\begin{aligned} G_1 &\equiv A_1(\ulcorner G_1\urcorner, \ulcorner G_2\urcorner) \\ G_2 &\equiv A_2(\ulcorner G_1\urcorner, \ulcorner G_2\urcorner) \end{aligned}$$

as claimed. □

Here's an application. In its informal version, the Postcard Paradox (due to Phillip Jourdain) involves a postcard on one side of which is written "The sentence on the other side is false", while on the other side it says: The sentence on the other side is true. More simply, it involves two sentences that refer to each other:

P1 Sentence (P2) is false.

P2 Sentence (P1) is true.

[77] If we are working in a less expressive language, then we can use the indirection discussed earlier.

We can formalize this paradox with the aid of the Generalized Diagonal Lemma. Let $A_1(x, y)$ be $\neg\mathsf{Tr}(y)$ and $A_2(x, y)$ be $\mathsf{Tr}(x)$. Then we have:

$$\begin{aligned} P_1 &\equiv \neg\mathsf{Tr}(\ulcorner P_2 \urcorner) \\ P_2 &\equiv \mathsf{Tr}(\ulcorner P_1 \urcorner) \end{aligned}$$

Longer such cycles can of course be constructed as well.[78]

4 'Direct' Self-Reference

There is an even stronger form of the Diagonal Lemma we might want to have.

Lemma (Directly Self-Referential Diagonal Lemma)**.** *For each formula $A(x)$, there is a* **numeral** $\overline{n_A}$ *such that* $\overline{n_A} = \ulcorner A(\overline{n_A}) \urcorner$.

There are papers in the literature that assume the Diagonal Lemma in this form (e.g. Rosenblatt, 2017, p. 97). It's convenient, if one has it, because Leibniz's Law is no longer required to show that $A(\overline{n_A})$ is equivalent (in whatever sense) to $A(\ulcorner A(\overline{n_A}) \urcorner)$. If we have the Directly Self-Referential Diagonal Lemma, then those are *the very same sentence*: The symbol $\ulcorner A(\overline{n_A}) \urcorner$ denotes the numeral for the Gödel number of $A(\overline{n_A})$; by hypothesis, that number is n_A; its numeral is called $\overline{n_A}$; so $\ulcorner A(\overline{n_A}) \urcorner$ and $\overline{n_A}$ are names of the same symbol. So $A(\overline{n_A})$ and $A(\ulcorner A(\overline{n_A}) \urcorner)$ are the same sentence.

The first point worth noting, then, is that this is *not* what even the Strong Diagonal Lemma delivers. That gives us a *term* t_A such that $t_A = \ulcorner A(t_A) \urcorner$, but that term (on the usual construction) is $\mathsf{diag}(\ulcorner A(\mathsf{diag}(x) \urcorner)$, which is obviously not a numeral. Indeed, not only do none of the Gödel numberings usually discussed in the literature allow us to prove the

[78] It's sometimes suggested that these two paradoxes show that self-reference is not really essential to the Liar Paradox. And, strictly speaking, that is surely true. But all the Postcard Paradox really shows is that self-reference wasn't the right notion: Circular chains of reference are the real problem; self-reference is just the simplest way of achieving such a chain. Yablo's Paradox poses a quite different challenge, but it remains at least somewhat controversial whether some kind of self-reference is present in Yablo's Paradox (see Cook (2014, ch. 2) for discussion and references). But even it if isn't (and I think it isn't), one might take the new lesson just to be that non-well-founded chains of reference are the real problem: Circular chains are just the simplest version of that. Indeed, that is what I have always thought Yablo's point was. (So, one might conclude, something like Kripke's diagnosis of the Liar, in terms of ungroundedness, is correct.)

Directly Self-Referential Diagonal Lemma, but they actually prohibit it. That is because most Gödel numberings have a property known as 'regularity' or 'monotonicity':[79] The Gödel number of any proper part of an expression is always smaller than the Gödel number of the whole expression. It follows from this that the Gödel number of a numeral must be at least as great as the number it names: $\ulcorner\overline{n}\urcorner \geq n$.[80] Moreover, the Gödel number of $A(\overline{n_A})$ must be greater than that of the numeral $\overline{n_A}$. So $\ulcorner A(\overline{n_A})\urcorner > \ulcorner\overline{n_A}\urcorner \geq n_A$, contrary to what the Directly Self-Referential Diagonal Lemma would require.[81]

We can, nonetheless, arrange for this kind of direct self-reference. There are, in fact, several different ways to do so.

The first, and simplest, is to use a Gödel numbering custom-built for the purpose. Here's one simple way to do that (Visser, 2004; Heck, 2007). Start with your favorite numbering; let $A_0(x), A_1(x), \ldots$ be an enumeration of the formulas with just x free. For each formula $A_i(x)$ in the mentioned list, let the new Gödel number of $A_i(\overline{2i+1})$ be $2i+1$; for every formula not of the form $A_i(\overline{2i+1})$, let its new Gödel number be twice its old one. By construction, then, for each formula $A_i(x)$, we will have: $\overline{2i+1} = \ulcorner A_i(\overline{2i+1})\urcorner$.[82]

This construction violates monotonicity. The importance of that constraint is debatable, but, even if one accepts it, there is another way to proceed that respects monotonicity. This involves the use of so-called 'efficient numerals', which are akin to binary numerals, or 'bicimals'. Efficient numerals play an important role in the study of weak systems of arithmetic, because their length does not grow as fast as does that of standard numerals.[83] For much the same reason, the short argument

[79] Sometimes this condition is made serious use of: See e.g. Łełyk (2022, p. 4).

[80] By induction. $\ulcorner 0\urcorner \geq 0$, trivially; so $\ulcorner S0\urcorner > \ulcorner 0\urcorner \geq 0$, so $\ulcorner S0\urcorner \geq 1$; and so forth. (Here, I'm using $\ulcorner A\urcorner$ to mean: the Gödel number of A.)

[81] If one thinks of numerals as an analogue of quote names, then one can see why this is unsurprising. (But see Grabmayr and Visser (2021, §6).) No quote name can be a name of an expression that contains that very quote name. That would require the quote name to be longer than itself. It's something like this thought, it seems to me, that makes people worry about the coherence of self-reference, or to think that it is irremediably circular. What such a worry overlooks is that there can be other ways of referring to sentences besides quote names.

[82] As a referee pointed out, this construction may not give each sentence a unique Gödel number, which is usually desirable (though not essential). One can resolve this issue by simply taking the numbering described and letting the 'real' Gödel number of a sentence be the least one. Note also that, if the original numbering was computable, so will be the new one; similarly for the inverses.

[83] The problem comes when one tries to define the function that maps a number to the

above that shows that monotonicity implies that $\ulcorner \overline{n} \urcorner \geq n$ does not hold for efficient numerals. A great deal more work is needed to produce a monotonic Gödel numbering that delivers the Directly Self-Referential Diagonal Lemma, but it is possible (Grabmayr and Visser, 2021, §4).[84]

Yet another strategy traces to Kripke's "Outline of a Theory of Truth". He writes:

> Let 'Jack' be a name of the sentence 'Jack is short', and we have a sentence that says of itself that it is short. I can see nothing wrong with "direct" self-reference of this type. If 'Jack' is not already a name in the language, why can we not introduce it as a name of any entity we please? In particular, why can it not be a name of the (uninterpreted) finite sequence of marks 'Jack is short'? (Would it be permissible to call this sequence of marks "Harry," but not "Jack"? Surely prohibitions on naming are arbitrary here.) There is no vicious circle in our procedure, since we need not interpret the sequence of marks 'Jack is short' before we name it. Yet if we name it "Jack", it at once becomes meaningful and true. (Kripke, 1975, p. 693)

This is by far the closest analogue of the technique I used, at the beginning of this chapter, to introduce a self-referential sentence:

(L) Sentence (L) is not true.

Here, "(L)" is supposed straightforwardly to be a *name* of the sentence displayed.

This being philosophy, there's more to be said about whether Kripke's strategy is enough to vindicate 'direct' self-reference (Visser, 2004, pp. 156ff). But a corresponding formal construction is not complicated. Let $A_0, A_1, \ldots$ again be an enumeration of the formulas with just x free; let $s_0, s_1, \ldots$ be an infinite sequence of (as yet uninterpreted) new constants; add these to our base language, and redo the Gödel numbering so that formulas involving them are included. Here's how we interpret the constants: Let the constant s_i denote the Gödel number of the formula $A_i(s_i)$; if we wish, we can add axioms of the form $s_i = \ulcorner A_i(s_i) \urcorner$ to

Gödel number of its numeral. Using typical Gödel numberings, and standard numerals, this function will have exponential growth (Grabmayr and Visser, 2021, Remark 2.4).

[84] For additional discussion of monotonicity, see Grabmayr et al. (2023).

whatever theory we were discussing.[85] These new axioms will assert, in effect, the various instances of the Directly Self-Referential Diagonal Lemma, though the new constants now play the role previously played by numerals.[86] No proof is now required of the Directly Self-Referential Diagonal Lemma, because it has been 'built into' the coding. (This is also true of the other two constructions mentioned above.)

Once again, it's important to note that just adding new constants is not what makes the Directly Self-Referential Diagonal Lemma available: One has also to arrange for the denotations of the new constants to be exactly what they need to be. This fact gives rise to a worry about these sorts of techniques.

The worry is in the vicinity of ones concerning the 'intensionality' of the Second Incompleteness Theorem. The proof of the First Incompleteness Theorem[87] requires only an 'extensionally correct' provability predicate. More precisely, it requires only that the formula $\mathsf{Bew}(x, y)$ should represent the proof-of relation in whatever theory Σ we are discussing. I.e., so long as we have:

1. $\Sigma \vdash \mathsf{Bew}(\overline{n}, \overline{m})$, if n codes a proof of the formula coded by m
2. $\Sigma \vdash \neg\mathsf{Bew}(\overline{n}, \overline{m})$, if n does not code such a proof

we will be able to mimic Gödel's reasoning. The Diagonal Lemma will deliver a sentence G such that $G \equiv \neg\exists x \mathsf{Bew}(x, \ulcorner G \urcorner)$; if Σ is ω-consistent, then Σ will neither prove nor refute G. It does not matter at all whether $\mathsf{Bew}(x, y)$ in any sense 'means' that x codes a proof of y.

In the case of the Second Incompleteness Theorem,[88] however, it is not enough for $\mathsf{Bew}(x, y)$ to represent the proof-of relation. The simplest counterexample is the carefully constructed provability predicate that Barkley Rosser (1936) uses to prove his strengthening of the First Incom-

[85] If one is interested only in producing a liar sentence, then one can introduce just one new term, say ℓ, redo the Gödel numbering to include it, and then let ℓ denote the Gödel number of $\neg\mathsf{Tr}(\ell)$.

[86] This construction seems to have been discovered independently by several people, reflecting on Kripke's discussion. Kripke (2021) himself recently published a short note describing it. Kripke also shows how this technique can be used to prove the First Incompleteness Theorem (p. 4). (Note, by the way, that it is irrelevant, in Kripke's proof, that S' is a *conservative* extension of S. So long as S' is any kind of extension of S, if S' does not prove A, then S does not prove it, either.)

[87] Which says that no sufficiently strong, consistent formal theory is complete.

[88] Which says that no sufficiently strong, consistent formal theory proves its own consistency.

pleteness Theorem. That formula, $\mathsf{RBew}(x,y)$, *also* represents the proof-of relation,[89] and yet it is absolutely trivial that $\neg\exists x\mathsf{RBew}(x,\ulcorner 0=1\urcorner)$. That is, it is easy to prove the 'Rosser consistency' of almost any theory you like, in that very theory (and, in fact, in Q). Solomon Feferman (1960) shows that there are much more interesting ways to get a theory seemingly to prove its own consistency, and there is now a large literature on exactly what it means for some statement to 'express' the consistency of a given theory.

As the reference to Feferman will have made clear, the literature on the intensionality of the Second Incompleteness Theorem is old and venerable. The corresponding issues for the semantic paradoxes, by contrast, have only begun to be discussed recently.[90] We saw an example of this kind of intensionality above: A theory whose consistency depends upon the Gödel numbering used. By contrast, Tarski's Theorem is largely independent of the Gödel numbering employed. As we have seen, there are some restrictions, but so long as the Gödel numbering meets some pretty weak conditions,[91] the set of Gödel numbers of true sentences will not be definable. That makes blaming the Liar Paradox on self-reference implausible: There are lots of easy ways to achieve self-reference, and all of them allow one to formulate the Liar Paradox. By contrast, suppose there is some 'paradox' whose derivation depends essentially upon the use of direct self-reference, in the sense we have been discussing here. Then it might not be so clear whether we have a genuine paradox. Maybe *that* kind of self-reference is indeed suspicious.

Something in this vicinity occurs in the literature on the so-called V-Curry (Beall and Murzi, 2013). Much of the interest of this paradox lies in its allegedly requiring for its derivation—besides certain 'naïve' rules for a predicate $\mathsf{Val}(x,y)$ expressing the validity of the inference from x to y—only so-called structural rules.[92] But that is true only if we

[89] This is what makes Rosser's strenghtening work: Once it's been shown that $\mathsf{RBew}(x,y)$ represents the proof-of relation, we can mimic Gödel's reasoning. But now we need only assume that Σ is consistent, not that it is ω-consistent.

[90] See, for example, Heck (2007); Grabmayr and Visser (2021); Halbach and Leigh (2022, Ch. 12). This literature intersects significantly with the emerging literature on the nature of self-reference: There are difficult questions about what exactly it means for a formula to '*say* of itself' that it has some property or other; see Halbach and Visser (2014a,b); Picollo (2018).

[91] It's enough for the image of the concatenation function to be definable in whatever theory is in question.

[92] In particular, the rule of contraction, which asserts (in effect) that the premises of an inference form a set, so that, if $A, A \vdash B$, then $A \vdash B$.

have the Directly Self-Referential Diagonal Lemma (Christman, 2023, Ch. 1). If we have only the Strong Diagonal Lemma, then we need to use Leibniz's Law to get the inter-derivability claim; as I emphasized above, that is the kind of principle that non-classical solutions to the semantic paradoxes have always questioned. So should we think that there are non-structural solutions to the V-Curry or not?[93] It depends upon what forms of self-reference one thinks are legitimate.[94]

References

Austin, J. (1950). 'Truth', *Proceedings of the Aristotelian Society* 24: 111–28. Reprinted in Austin, 1961, Ch. 5.

—— (1961). *Philosophical Papers*, Urmson, J. and Warnock, G., eds. Oxford, Oxford University Press.

Beall, J. and Murzi, J. (2013). 'Two flavors of Curry's paradox', *Journal of Philosophy* 110: 143–165.

Boolos, G. (1993). *The Logic of Provability*. New York, Cambridge University Press.

—— (1998). 'Quotational ambiguity', in R. Jeffrey (ed.), *Logic, Logic, and Logic*. Cambridge MA, Harvard University Press, 392–405.

Boolos, G. S., Burgess, J. P., and Jeffrey, R. C. (2007). *Computability and Logic*, 5th edition. Cambridge, Cambridge University Press.

Burge, T. (1984). 'Semantical paradox', in Martin 1984, 83–118.

Carnap, R. (1937). *The Logical Syntax of Language*. London, Routledge.

Christman, M. (2023). *TBA*. PhD thesis, Brown University.

Cook, R. T. (2014). *The Yablo Paradox*. Oxford, Oxford University Press.

[93] With Field (2017), I'm inclined myself to doubt the principles governing $\mathsf{Val}(x, y)$ to which the derivation appeals, in which case the question is moot. But it can nonetheless be used to illustrate the point made in the text.

[94] Thanks to Albert Visser for several years now of productive conversations about self-reference and the incompleteness theorems, and to Mark Christman: Some of the ideas in this paper emerged in conversation with him. Thanks also to an anonymous referee for some *really* helpful comments. Finally, thanks to the editors for the invitation to contribute to this volume.

Corcoran, J., Frank, W., and Maloney, M. (1974). 'String theory', *Journal of Symbolic Logic* 39: 625–37.

Davidson, D. (1984). 'Quotation', in *Inquiries Into Truth and Interpretation*. Oxford, Clarendon Press, 79–92.

Dedekind, R. (1902). 'The nature and meaning of numbers', tr. by W. W. Beman, in *Essays on the theory of numbers*. Chicago, The Open Court Publishing Company, 31–115.

Feferman, S. (1960). 'Arithmetization of metamathematics in a general setting', *Fundamenta Mathematicae* 49: 35–92.

Field, H. (2017). 'Disarming a paradox of validity', *Notre Dame Journal of Formal Logic* 58: 1–19.

Frege, G. (1879). *Begriffsschrift, eine der arithmetischen nachgebildete Formelsprache des reinen Denkens*. Louis Nebert. Translated as Frege, 1967.

—— (1967). 'Begriffsschrift: A formula language modeled upon that of arithmetic, for pure thought', tr. by S. Bauer-Mengelberg, in J. van Heijenoort (ed.), *From Frege to Gödel: A Sourcebook in Mathematical Logic 1879–1931*. Cambridge MA, Harvard University Press, 5–82.

Gaifman, H. (1992). 'Pointers to truth', *Journal of Philosophy* 89: 223–61.

—— (2006). 'Naming and diagonalization, from Cantor to Gödel to Kleene', *Logic Journal of the IGPL* 14: 709–28.

Glanzberg, M. (2001). 'The liar in context', *Philosophical Studies* 103: 217–51.

Gödel, K. (1931). 'Über formal unentscheidbare Sätze der *Principia Mathematica* und verwandter Systeme, I.', *Monatshefte für Mathematik und Physik* 38: 173–98. Translated as Gödel, 1986.

—— (1986). 'On formally undecidable propositions of *Principia Mathematica* and related systems I', in S. Feferman, *et al*. (eds.), *Collected Works*, volume 1, 3d edition. Oxford, Oxford University Press, 144–95.

Goldstein, L. (1992). "This statement is not true' is not true', *Analysis* 52: 1–5.

Grabmayr, B. (2021). ‘On the invariance of Gödel’s second theorem with regard to numberings’, *Review of Symbolic Logic* 14: 51–84.

Grabmayr, B., Halbach, V., and Ye, L. (2023). ‘Varieties of self-reference in metamathematics’, *Journal of Philosophical Logic* 52: 1005–52.

Grabmayr, B. and Visser, A. (2021). ‘Self-reference upfront: A study of self-referential Gödel numberings’, *The Review of Symbolic Logic*. Forthcoming.

Grzegorczyk, A. (2005). ‘Undecidability without arithmetization’, *Studia Logica* 79: 163–230.

Hájek, P. and Pudlák, P. (1993). *Metamathematics of First-order Arithmetic*. New York, Springer-Verlag.

Halbach, V. and Leigh, G. E. (2022). *The Road to Paradox: A Guide to Syntax, Truth, and Modality*. Cambridge, Cambridge University Press. Forthcoming.

Halbach, V. and Visser, A. (2014a). ‘Self-reference in arithmetic I’, *The Review of Symbolic Logic* 7: 671–91.

—— (2014b). ‘Self-reference in arithmetic II’, *The Review of Symbolic Logic* 7: 692–712.

Halbach, V. and Zhang, S. (2017). ‘Yablo without Gödel’, *Analysis* 77: 53–9.

Heck, R. K. (2007). ‘Self-reference and the languages of arithmetic’, *Philosophia Mathematica* 15: 1–29. Originally published under the name “Richard G. Heck, Jr”.

—— (2014). ‘Predicative Frege arithmetic and “everyday mathematics”’, *Philosophia Mathematica* 22: 279–307. Originally published under the name “Richard G. Heck, Jr”.

—— (2022). ‘Formal background for the incompleteness and undefinability theorems’. `https://philpapers.org/archive/HECFBF.pdf`.

—— (2023). ‘Disquotation, translation, and context-dependence’, in Lepore and Sosa 2023, 99–128.

Jeroslow, R. G. (1973). 'Redundancies in the Hilbert-Bernays derivability conditions for Gödel's second incompleteness theorem', *Journal of Symbolic Logic* 38: 359–67.

Ketland, J. (2005). 'Yablo's Paradox and ω-inconsistency', *Synthese* 145: 295–302.

Kripke, S. (1975). 'Outline of a theory of truth', *Journal of Philosophy* 72: 690–716.

—— (2021). 'Gödel's theorem and direct self-reference', *The Review of Symbolic Logic*. Forthcoming.

Łełyk, M. Z. (2022). 'Model theory and proof theory of the global reflection principle', *The Journal of Symbolic Logic* 88: 738–79.

Lepore, E. and Sosa, D., eds. (2023). *Oxford Studies in Philosophy of Language*, volume 3. Oxford, Oxford University Press.

Martin, R., ed. (1984). *Recent Essays on the Liar Paradox*. New York, Oxford University Press.

Milne, P. (2007). 'On Gödel sentences and what they say', *Philosophia Mathematica* 15: 193–226.

Mostowski, A. (1952). *Sentences Undecidable in Formalized Arithmetic: An Exposition of the Theory of Kurt Gödel*. Amsterdam, North-Holland Publishing Company.

Murzi, J. and Rossi, L. (2017). 'Naïve validity', *Synthese* 199: S819–41.

Parsons, C. (1981). 'The liar paradox', in *Mathematics in Philosophy*. Ithaca NY, Cornell University Press, 221–67.

Picollo, L. (2018). 'Reference in arithmetic', *Review of Symbolic Logic* 11: 573–603.

Priest, G. (1997). 'Yablo's paradox', *Analysis* 57: 236–42.

Quine, W. V. O. (1946). 'Concatenation as a basis for arithmetic', *Journal of Symbolic Logic* 11: 105–14.

—— (1981). *Mathematical Logic*, revised edition. Cambridge MA, Harvard University Press.

Rescorla, M. (2007). 'Church's thesis and the conceptual analysis of computability', *Notre Dame Journal of Formal Logic* 48: 253–80.

Rosenblatt, L. (2017). 'Naive validity, internalization, and substructural approaches to paradox', *Ergo: An Open Access Journal of Philosophy* 4: 93–120.

Rosser, B. (1936). 'Extensions of some theorems of Gödel and Church', *Journal of Symbolic Logic* 1: 87–91.

—— (1939). 'An informal exposition of proofs of gödel's theorems and Church's theorem', *Journal of Symbolic Logic* 4: 53–60.

Russell, B. (1905). 'On denoting', *Mind* 14: 479–93.

Simmons, K. (1993). *Universality and the Liar: An Essay on Truth and the Diagonal Argument*. Cambridge, Cambridge University Press.

Skyrms, B. (1984). 'Intensional aspects of semantical self-reference', in Martin 1984, 119–31.

Smith, P. (2013). *An Introduction to Gödel's Incompleteness Theorems*, 2d edition. Cambridge, Cambridge University Press.

Smoryński, C. (1981). 'Fifty years of self-reference in arithmetic', *Notre Dame Journal of Formal Logic* 22: 357–74.

Smullyan, R. M. (1957). 'Languages in which self-reference is possible', *Journal of Symbolic Logic* 22: 55–67.

—— (1982). *The Gödel Incompleteness Theorems*. Oxford, Oxford University Press.

—— (1994). *Diagonalization and Self-Referece*. Oxford, Clarendon Press.

Švejdar, V. (2007). 'An interpretation of Robinson arithmetic in Grzegorczyk's weaker variant', *Fundamenta Informaticae* 81: 347–54.

Tarski, A. (1956). 'The concept of truth in formalized languages', in J. Corcoran (ed.), *Logic, Semantics, and Metamathematics*. Indianapolis, Hackett, 152–278.

Tarski, A., Mostowski, A., and Robinson, A. (1953). *Undecidable Theories*. Amsterdam, North-Holland Publishing.

Visser, A. (1998). ‘An overview of interpretability logic’, in M. Kracht, *et al*. (eds.), *Advances in Modal Logic*, volume 1. Stanford, CSLI Publications, 307–59.

—— (2004). ‘Semantics and the liar paradox’, in D. Gabbay (ed.), *Handbook of Philosophical Logic*, volume 11, 2d edition. Dordrecht, Kluwer Academic Publishers.

—— (2009). ‘Growing commas: A study of sequentiality and concatenation’, *Notre Dame Journal of Formal Logic* 50: 61–85.

Whitehead, A. N. and Russell, B. (1925). *Principia Mathematica*, 2d edition, volume 1. Cambridge, Cambridge University Press.

Yablo, S. (1993). ‘Paradox without self-reference’, *Analysis* 53: 251–2.