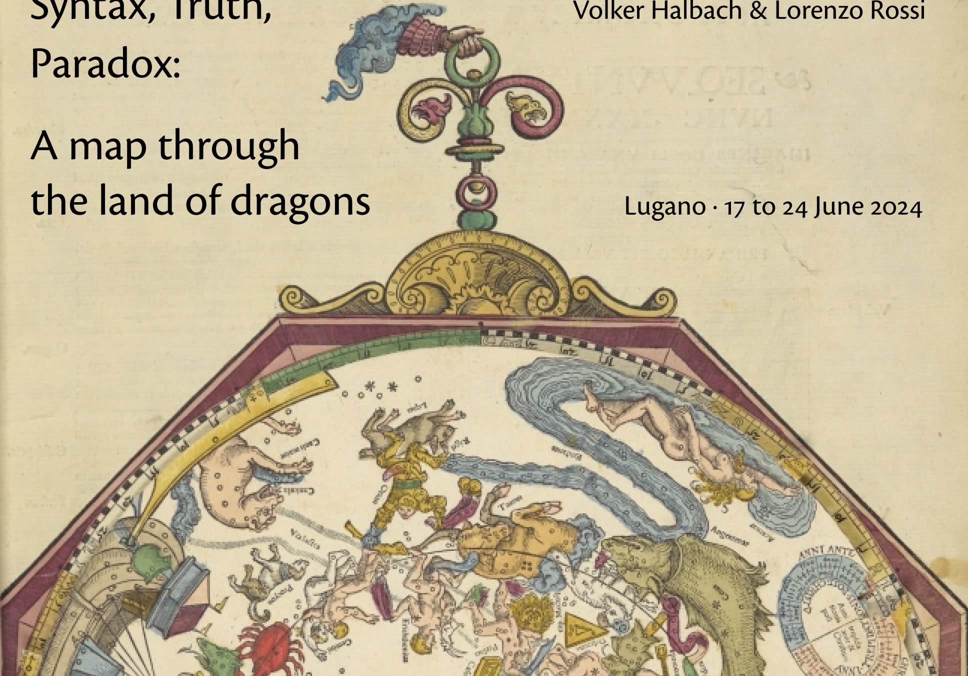
Syntax, Truth, Paradox:

A map through the land of dragons

Volker Halbach & Lorenzo Rossi

Lugano · 17 to 24 June 2024



Prelude: The paradoxes in philosophy

Day 1: A theory of expressions

Day 2: The paradoxes

Day 3: Possible-worlds analysis of the paradoxes

Day 4: Truth!

A THEORY OF EXPRESSIONS

Content

Ontology

I opt for the predicate approach. But of which objects should we predicate truth, necessity, provability, analyticity, and so on?

(i) Sentences
  (a) sentence tokens or types
  (b) sentences interpreted or uninterpreted
  (c) sentences identified with numbers or sets
(ii) Propositions
  (a) coarse- or fine-grained
  (b) propositions as sets of possible worlds (which are themselves not members of any world)
  (c) propositions as language independent or not (usually not nowadays

There are also beliefs, states of affairs, facts, judgements, and so on.

I opt for the predicate approach. But of which objects should we predicate truth, necessity, provability, analyticity, and so on?

(i) Sentences
   (a) sentence tokens or types
   (b) sentences interpreted or uninterpreted
   (c) sentences identified with numbers or sets
(ii) Propositions
   (a) coarse- or fine-grained
   (b) propositions as sets of possible worlds (which are themselves not members of any world)
   (c) propositions as language independent or not (usually not nowadays

There are also beliefs, states of affairs, facts, judgements, and so on.

The canonical answers are probably:

Propositions are true, known, and necessary. The propositions that are true or necessary tend to be coarse grained in the literature, while those that are believed are often assumed to be more fine grained.

Sentences are provable and analytic. They are usually taken to be sentences types.

Problem:

What is necessary or known is true. Analytic truths are necessary.

One single kind of objects is needed.

The canonical answers are probably:

Propositions are true, known, and necessary. The propositions that are true or necessary tend to be coarse grained in the literature, while those that are believed are often assumed to be more fine grained.

Sentences are provable and analytic. They are usually taken to be sentences types.

Problem:

What is necessary or known is true. Analytic truths are necessary.

One single kind of objects is needed.

THE HENRY FORD THEORY OF PROPOSITIONS

You can have any kind of truth (necessity etc.)bearers as long as they share their structures with sentences types.

Quine (1970), who denounced propositions as *creatures of darkness*, suggested a reduction, if you don't like sentences as truth bearers: Instead of saying of a sentence *S*, '*S* is true' you can say: '*S* expresses a true proposition.'

At any rate we access propositions via sentences.

The Henry Ford theory of propositions

You can have any kind of truth (necessity etc.)bearers as long as they share their structures with sentences types.

Quine (1970), who denounced propositions as *creatures of darkness*, suggested a reduction, if you don't like sentences as truth bearers: Instead of saying of a sentence *S*, '*S* is true' you can say: '*S* expresses a true proposition.'

At any rate we access propositions via sentences.

I attribute truth, necessity, and so on to *types of sentences*.

I need a theory of sentences and their constituents.

There are already many theories of syntax. Here are my desiderata for such a theory of syntax:

(i) It should axiomatize our informal reasoning about formal languages in a natural way (unlike 'elegant' theories).
(ii) It should mainly be a theory *of its own syntax*, not like like Tarski's theory in the 'Concept of Truth'.
(iii) It should not add 'extra structure' as theories via coding do.

Syntactic objects have a strange life between the realm of concrete objects and the realm of the abstract objects.

I attribute truth, necessity, and so on to *types of sentences*.

I need a theory of sentences and their constituents.

There are already many theories of syntax. Here are my desiderata for such a theory of syntax:

(i) It should axiomatize our informal reasoning about formal languages in a natural way (unlike 'elegant' theories).
(ii) It should mainly be a theory *of its own syntax*, not like like Tarski's theory in the 'Concept of Truth'.
(iii) It should not add 'extra structure' as theories via coding do.

Syntactic objects have a strange life between the realm of concrete objects and the realm of the abstract objects.

I attribute truth, necessity, and so on to *types of sentences*.

I need a theory of sentences and their constituents.

There are already many theories of syntax. Here are my desiderata for such a theory of syntax:

(i) It should axiomatize our informal reasoning about formal languages in a natural way (unlike 'elegant' theories).
(ii) It should mainly be a theory *of its own syntax*, not like like Tarski's theory in the 'Concept of Truth'.
(iii) It should not add 'extra structure' as theories via coding do.

Syntactic objects have a strange life between the realm of concrete objects and the realm of the abstract objects.

I attribute truth, necessity, and so on to *types of sentences*.

I need a theory of sentences and their constituents.

There are already many theories of syntax. Here are my desiderata for such a theory of syntax:

(i) It should axiomatize our informal reasoning about formal languages in a natural way (unlike 'elegant' theories).
(ii) It should mainly be a theory *of its own syntax*, not like like Tarski's theory in the 'Concept of Truth'.
(iii) It should not add 'extra structure' as theories via coding do.

Syntactic objects have a strange life between the realm of concrete objects and the realm of the abstract objects.

Of course, sentence types are abstract; they are not stains of ink on paper, or a bunch of diodes flashing up. But there are different degrees of abstraction.

(i) Is ‘W’ the same type as ‘W’ or ‘w’ or ‘W’ etc.?
Of course I would like to abstract away from serif vs sans-serif etc.

(ii) If ‘V’ a subexpression of ‘W’ ? Is ‘W’ a composed expression? Is ‘V’ composed?
I would like both, ‘W’ and ‘V’ to be atomic. I don’t case about the graphic shape, as long as the letters are distinct. ‘VV’ is composed of two atomic expressions, while ‘W’ is not.
What makes an expression atomic? Is ‘i’ atomic?

(iii) Should we think of complex expressions – and sentences, in particular – as linear strings of objects, as sequences of sequences or as trees?

(iv) Form expressions of a fixed language only a free algebra?

And I still completely ignore spoken languages.

Of course, sentence types are abstract; they are not stains of ink on paper, or a bunch of diodes flashing up. But there are different degrees of abstraction.

(i) Is 'W' the same type as 'W' or 'w' or 'W' etc.?
Of course I would like to abstract away from serif vs sans-serif etc.

(ii) If 'V' a subexpression of 'W' ? Is 'W' a composed expression? Is 'V' composed?
I would like both, 'W' and 'V' to be atomic. I don't case about the graphic shape, as long as the letters are distinct. 'VV' is composed of two atomic expressions, while 'W' is not.
What makes an expression atomic? Is 'i' atomic?

(iii) Should we think of complex expressions – and sentences, in particular – as linear strings of objects, as sequences of sequences or as trees?

(iv) Form expressions of a fixed language only a free algebra?

And I still completely ignore spoken languages.

Of course, sentence types are abstract; they are not stains of ink on paper, or a bunch of diodes flashing up. But there are different degrees of abstraction.

(i) Is ‘W’ the same type as ‘W’ or ‘w’ or ‘W’ etc.?
Of course I would like to abstract away from serif vs sans-serif etc.

(ii) If ‘V’ a subexpression of ‘W’ ? Is ‘W’ a composed expression? Is ‘V’ composed?
I would like both, ‘W’ and ‘V’ to be atomic. I don’t case about the graphic shape, as long as the letters are distinct. ‘VV’ is composed of two atomic expressions, while ‘W’ is not.
What makes an expression atomic? Is ‘i’ atomic?

(iii) Should we think of complex expressions – and sentences, in particular – as linear strings of objects, as sequences of sequences or as trees?

(iv) Form expressions of a fixed language only a free algebra?

And I still completely ignore spoken languages.

Of course, sentence types are abstract; they are not stains of ink on paper, or a bunch of diodes flashing up. But there are different degrees of abstraction.

(i) Is ‘W’ the same type as ‘W’ or ‘w’ or ‘W’ etc.?
Of course I would like to abstract away from serif vs sans-serif etc.

(ii) If ‘V’ a subexpression of ‘W’ ? Is ‘W’ a composed expression? Is ‘V’ composed?
I would like both, ‘W’ and ‘V’ to be atomic. I don’t case about the graphic shape, as long as the letters are distinct. ‘VV’ is composed of two atomic expressions, while ‘W’ is not.
What makes an expression atomic? Is ‘i’ atomic?

(iii) Should we think of complex expressions – and sentences, in particular – as linear strings of objects, as sequences of sequences or as trees?

(iv) Form expressions of a fixed language only a free algebra?

And I still completely ignore spoken languages.

Of course, sentence types are abstract; they are not stains of ink on paper, or a bunch of diodes flashing up. But there are different degrees of abstraction.

(i) Is ‘W’ the same type as ‘W’ or ‘w’ or ‘W’ etc.?
Of course I would like to abstract away from serif vs sans-serif etc.

(ii) If ‘V’ a subexpression of ‘W’ ? Is ‘W’ a composed expression? Is ‘V’ composed?
I would like both, ‘W’ and ‘V’ to be atomic. I don’t case about the graphic shape, as long as the letters are distinct. ‘VV’ is composed of two atomic expressions, while ‘W’ is not.
What makes an expression atomic? Is ‘i’ atomic?

(iii) Should we think of complex expressions – and sentences, in particular – as linear strings of objects, as sequences of sequences or as trees?

(iv) Form expressions of a fixed language only a free algebra?

And I still completely ignore spoken languages.

These are serious metaphysical questions. I believe that propositions are creatures of darkness, but am not sure that sentences are creatures of light.

I need to proceed under certain assumptions. An atomic symbol cannot be obtained by composing other symbols (cf. ‘W’ and ‘V’). How exactly this is achieved is not my problem.

I understand expressions as strings of symbols (obtained by concatenation).

With only one symbol | things are easier. Arithmetic can be seen as the theory of syntax of a language with only one symbol.

These are serious metaphysical questions. I believe that propositions are creatures of darkness, but am not sure that sentences are creatures of light.

I need to proceed under certain assumptions. An atomic symbol cannot be obtained by composing other symbols (cf. 'W' and 'V'). How exactly this is achieved is not my problem.

I understand expressions as strings of symbols (obtained by concatenation).

With only one symbol | things are easier. Arithmetic can be seen as the theory of syntax of a language with only one symbol.

These are serious metaphysical questions. I believe that propositions are creatures of darkness, but am not sure that sentences are creatures of light.

I need to proceed under certain assumptions. An atomic symbol cannot be obtained by composing other symbols (cf. ‘W’ and ‘V’). How exactly this is achieved is not my problem.

I understand expressions as strings of symbols (obtained by concatenation).

With only one symbol | things are easier. Arithmetic can be seen as the theory of syntax of a language with only one symbol.

Definition

The symbols of $\mathcal{L}$ are:

(i) infinitely many variable symbols $\mathsf{v}_0, \mathsf{v}_1, \mathsf{v}_2, \mathsf{v}_3, \ldots$,

(ii) predicate symbols $=$ and $\square$,

(iii) function symbols q, $^\frown$, and sub,

(iv) the connectives $\neg$, $\rightarrow$ and the quantifier symbol $\forall$,

(v) auxiliary symbols (and),

(vi) possibly finitely many further function and predicate symbols of arbitrary arities and finitely many further auxiliary symbols, and

(vii) for each string e of symbols exactly one constant.

The language $\mathcal{L}$

DEFINITION CONTINUED

All symbols are pairwise distinct. For instance, v_0 is distinct from $\mathsf{v}_1, \mathsf{v}_2, \ldots, =$, and so on; v_1 is distinct from $\mathsf{v}_2, \ldots, =$, and so on. In particular, if e is any string of symbols, the constant for e is distinct from e itself and from all symbols in (i)–(vi); and if f is a string of symbols distinct from e, then the constants for e and f are also distinct. Consequently, the constant for e is distinct from the constant for the *constant* of e, and so on. For (vii) we assume that each constant is also associated with an expression, although this will be needed only later.

There are no further symbols in $\mathcal{L}$ beyond those in (i)–(vii).

The problem mentioned above are eliminated by the following assumption:

UNIQUE READABILITY ASSUMPTION

Assume that $a_1, \ldots, a_n, b_1, \ldots, b_k$ are symbols of $\mathcal{L}$. If the string $a_1 \ldots a_n$ is identical to the string $b_1 \ldots b_k$, then $n = k$, $a_1 = b_1, \ldots$, and $a_n = b_k$.

Example outside $\mathcal{L}$: W and VV. The latter was generated by two V with reduced kerning.

The language $\mathcal{L}$ itself has different notations.

Quotation constants are strange because of this. In my notation they are complex and one can read off from then for which expression they are a constant. There are other notations.

The overlining notation has its own problems.

Later quotations will be understood as complex.

The language $\mathcal{L}$ itself has different notations.

Quotation constants are strange because of this. In my notation they are complex and one can read off from then for which expression they are a constant. There are other notations.

The overlining notation has its own problems.

Later quotations will be understood as complex.

Definition

The $\mathcal{L}$-terms are defined as follows:

(i) All variables are terms.

(ii) All quotation constants are terms.

(iii) If t, r, and s are terms, then $\mathsf{q}t$, $^\frown st$, and $\mathsf{sub}\, rst$ are terms.

(iv) If $t_1, \ldots, t_n$ are terms and f is one of the additional function symbols of arity n, then $ft_1 \ldots t_n$ is a term.

(v) Nothing else is an $\mathcal{L}$-term.

The term $^\frown st$ will be written as $(s^\frown t)$. $(s^\frown t^\frown u)$ is short for $\big((s^\frown t)^\frown u\big)$. We will also often add brackets and commas for readability and write, for instance, $\mathsf{sub}(r, s, t)$ instead of $\mathsf{sub}\, rst$. In the following definitions we drop the analogous clauses stating that nothing else is a formula, sentence, and so on.

$\bar{}$ is a term. I write $\underline{0}$ for $\bar{}$.

Definition

The atomic $\mathcal{L}$-formulæ are defined as follows:

(i) If s and t are terms, then $=st$ and $\Box s$ are atomic formulæ.

(ii) If $t_1, \ldots, t_n$ are terms and P is one of the additional predicate symbols of arity n, then $Pt_1 \ldots t_n$ is an atomic formula.

The atomic formula $=st$ is written as $s = t$.

Definition

If φ and ψ are formulæ and x is a variable, then $\neg\varphi$, $(\varphi \to \psi)$, and $\forall x\, \varphi$ are formulæ.

DEFINITION

The atomic $\mathcal{L}$-formulæ are defined as follows:

(i) If s and t are terms, then $=st$ and $\Box s$ are atomic formulæ.
(ii) If $t_1, \ldots, t_n$ are terms and P is one of the additional predicate symbols of arity n, then $Pt_1 \ldots t_n$ is an atomic formula.

The atomic formula $=st$ is written as $s = t$.

DEFINITION

If φ and ψ are formulæ and x is a variable, then $\neg\varphi$, $(\varphi \to \psi)$, and $\forall x\, \varphi$ are formulæ.

Definition

(i) Every occurrence of a variable in an atomic formula is free in that formula.

(ii) All occurrences of free variables y in φ are also free in $\forall x\, \varphi$ iff y is distinct from x. All other occurrences of variables are not free.

An occurrence of a variable in a formula is bound iff it is not free.

Definition

A formula is a sentence iff it does not contain a free occurrence of a variable.

Definition

(i) Every occurrence of a variable in an atomic formula is free in that formula.

(ii) All occurrences of free variables y in φ are also free in $\forall x\, \varphi$ iff y is distinct from x. All other occurrences of variables are not free.

An occurrence of a variable in a formula is bound iff it is not free.

Definition

A formula is a sentence iff it does not contain a free occurrence of a variable.

$$\neg\Big(\forall \mathsf{v}_3\Big(\mathsf{v}_3 = \overline{\rightarrow\forall} \;\rightarrow\; \neg\, \mathsf{v}_3 = \overline{\overline{\rightarrow\forall}}\,\Big) \;\rightarrow\; \Box\overline{\mathsf{v}_3}\Big),$$

$$\overline{\mathsf{v}_{12}} = \overline{\neg\Box\overline{\neg}}.$$

The axioms

I will now give the axioms of the theory E.

I will say that E contains certain axioms and rules, but it may also contain more. I aim at a minimal set of assumptions that are sufficient for generating the paradoxes. The weaker the assumption, the stronger the inconsistency result.

If we try to prove more fancy result, we have to make more assumptions about E.

Definition

All instances of the following schemas are axioms of E:

A1 $\overline{a} ^\frown \overline{b} = \overline{ab}$, where a and b are arbitrary strings of symbols

A2 $\mathsf{q}(\overline{a}) = \overline{\overline{a}}$

A3 $\mathsf{sub}(\overline{a}, \overline{b}, \overline{c}) = \overline{d}$, where a and c are arbitrary strings of symbols, b is a symbol (or, equivalently, a length-1 string of symbols), and d is the string of symbols obtained from a by replacing all occurrences of the symbol b with c

A4 $\forall \mathsf{x} \forall \mathsf{y} \forall \mathsf{z} \big((\mathsf{x} ^\frown \mathsf{y}) ^\frown \mathsf{z}\big) = \big(\mathsf{x} ^\frown (\mathsf{y} ^\frown \mathsf{z})\big)$

A5 $\forall \mathsf{x} \forall \mathsf{y} \big(\mathsf{x} ^\frown \mathsf{y} = \underline{0} \rightarrow \mathsf{x} = \underline{0} \wedge \mathsf{y} = \underline{0}\big)$

A6 $\forall \mathsf{x} \forall \mathsf{y} (\mathsf{x} ^\frown \mathsf{y} = \mathsf{x} \leftrightarrow \mathsf{y} = \underline{0}) \ \wedge \ \forall \mathsf{x} \forall \mathsf{y} (\mathsf{y} ^\frown \mathsf{x} = \mathsf{x} \leftrightarrow \mathsf{y} = \underline{0})$

A7 $\forall \mathsf{x} \forall \mathsf{y}\ \mathsf{sub}(\mathsf{x} ^\frown \overline{a}, \overline{a}, \mathsf{y}) = \mathsf{sub}(\mathsf{x}, \overline{a}, \mathsf{y}) ^\frown \mathsf{y}$, where a is a symbol

A8 $\forall \mathsf{x} \forall \mathsf{y} \forall \mathsf{z} \forall \mathsf{w}$
$\Big(\mathsf{x} ^\frown \mathsf{y} = \mathsf{z} ^\frown \mathsf{w} \leftrightarrow \exists \mathsf{v}_4 \big((\mathsf{x} = \mathsf{z} ^\frown \mathsf{v}_4 \wedge \mathsf{v}_4 ^\frown \mathsf{y} = \mathsf{w}) \vee (\mathsf{x} ^\frown \mathsf{v}_4 = \mathsf{z} \wedge \mathsf{y} = \mathsf{v}_4 ^\frown \mathsf{w})\big)\Big)$

I have added brackets to A2, A3, and A7 and used infix notation.

The concatenation of two expressions e_1 and e_2 is simply the expression e_1 followed by e_2. For instance, $\neg\neg\mathsf{v}$ is the concatenation of $\neg$ and $\neg\mathsf{v}$.

Therefore $\overline{\neg\neg\mathsf{v}} = \overline{\neg} \frown \overline{\neg\mathsf{v}}$ is an instance of A1 as well as $\overline{\neg\neg\mathsf{v}} = \overline{\neg\neg} \frown \overline{\mathsf{v}}$.

Concatenating the empty string with any expression e gives again the same expression e. Therefore we have, for instance, $\overline{\forall} \frown \underline{0} = \overline{\forall}$ as an instance of A1.

The concatenation of two expressions e_1 and e_2 is simply the expression e_1 followed by e_2. For instance, $\neg\neg\mathsf{v}$ is the concatenation of $\neg$ and $\neg\mathsf{v}$.

Therefore $\overline{\neg\neg\mathsf{v}} = \overline{\neg} \frown \overline{\neg\mathsf{v}}$ is an instance of A1 as well as $\overline{\neg\neg\mathsf{v}} = \overline{\neg\neg} \frown \overline{\mathsf{v}}$.

Concatenating the empty string with any expression e gives again the same expression e. Therefore we have, for instance, $\overline{\forall} \frown \underline{0} = \overline{\forall}$ as an instance of A1.

The concatenation of two expressions e_1 and e_2 is simply the expression e_1 followed by e_2. For instance, ¬¬v is the concatenation of ¬ and ¬v.

Therefore $\overline{\neg\neg\mathsf{v}} = \overline{\neg} \frown \overline{\neg\mathsf{v}}$ is an instance of A1 as well as $\overline{\neg\neg\mathsf{v}} = \overline{\neg\neg} \frown \overline{\mathsf{v}}$.

Concatenating the empty string with any expression e gives again the same expression e. Therefore we have, for instance, $\overline{\forall} \frown \underline{0} = \overline{\forall}$ as an instance of A1.

An instance of A2 is the sentence $\mathsf{q}\overline{\overline{\mathsf{v}}\neg} = \overline{\overline{\overline{\mathsf{v}}\neg}}$. Thus q describes the function that takes an expression and returns its quotation constant.

In A3 I have imposed the restriction that b must be a single symbol. This does not imply that the substitution function cannot be applied to complex expressions; just A3 does not say anything about the result of substituting a complex expression.

The reason for this restriction is that the result of substitution of a complex strings may be not unique. For instance, the result of substituting $\neg$ for $\wedge\wedge$ in $\wedge \wedge \wedge$ might be either $\wedge\neg$ or $\neg\wedge$. The problem can be fixed in several ways, but I do not need to substitute complex expressions in the following. Therefore I do not 'solve' the problem but avoid it by the restriction of b to a single symbol.

In A3 I have imposed the restriction that b must be a single symbol. This does not imply that the substitution function cannot be applied to complex expressions; just A3 does not say anything about the result of substituting a complex expression.

The reason for this restriction is that the result of substitution of a complex strings may be not unique. For instance, the result of substituting $\neg$ for $\wedge\wedge$ in $\wedge \wedge \wedge$ might be either $\wedge\neg$ or $\neg\wedge$. The problem can be fixed in several ways, but I do not need to substitute complex expressions in the following. Therefore I do not 'solve' the problem but avoid it by the restriction of b to a single symbol.

A1-A3 are already sufficient for proving the diagonalization Theorem 10.

A4 simplifies the reasoning with strings a great deal. Since $\mathsf{E} \vdash (x \frown y) \frown z = x \frown (y \frown z)$, that is, $\frown$ is associative by A4, I shall simply write $x \frown y \frown z$. for the sake of definiteness we can stipulate that $x \frown y \frown z$ is short for $(x \frown y) \frown z$ and similarly for more applications of $\frown$.

A1-A3 are already sufficient for proving the diagonalization Theorem 10.

A4 simplifies the reasoning with strings a great deal. Since $\mathsf{E} \vdash (x \frown y) \frown z = x \frown (y \frown z)$, that is, $\frown$ is associative by A4, I shall simply write $x \frown y \frown z$. for the sake of definiteness we can stipulate that $x \frown y \frown z$ is short for $(x \frown y) \frown z$ and similarly for more applications of $\frown$.

I write $\mathsf{E} \vdash \varphi$ if and only if the formula φ is a logical consequence of the theory E.

EXAMPLE

$$\mathsf{E} \vdash \mathsf{sub}(\overline{\neg\neg}, \overline{\neg}, \overline{\neg\neg\neg}) = \overline{\neg\neg\neg\neg\neg\neg}$$

EXAMPLE

$$\mathsf{E} \vdash \mathsf{sub}(\overline{\mathsf{v} = \mathsf{v} \wedge \overline{\mathsf{v}} = \overline{\mathsf{v}}}, \overline{\mathsf{v}}, \overline{\mathsf{v}_2}) = \overline{\mathsf{v}_2 = \mathsf{v}_2 \wedge \overline{\mathsf{v}} = \overline{\mathsf{v}}}$$

I write $\mathsf{E} \vdash \varphi$ if and only if the formula φ is a logical consequence of the theory E.

EXAMPLE

$\mathsf{E} \vdash \mathsf{sub}(\overline{\neg\neg}, \overline{\neg}, \overline{\neg\neg\neg}) = \overline{\neg\neg\neg\neg\neg\neg\neg}$

EXAMPLE

$\mathsf{E} \vdash \mathsf{sub}(\overline{\mathsf{v} = \mathsf{v} \wedge \overline{\mathsf{v}} = \overline{\mathsf{v}}}, \overline{\mathsf{v}}, \overline{\mathsf{v}_2}) = \overline{\mathsf{v}_2 = \mathsf{v}_2 \wedge \overline{\mathsf{v}} = \overline{\mathsf{v}}}$

I write $\mathsf{E} \vdash \varphi$ if and only if the formula φ is a logical consequence of the theory E.

EXAMPLE

$\mathsf{E} \vdash \mathsf{sub}(\overline{\neg\neg}, \overline{\neg}, \overline{\neg\neg\neg}) = \overline{\neg\neg\neg\neg\neg\neg\neg}$

EXAMPLE

$\mathsf{E} \vdash \mathsf{sub}(\overline{\mathsf{v} = \mathsf{v} \wedge \overline{\mathsf{v}} = \overline{\mathsf{v}}}, \overline{\mathsf{v}}, \overline{\mathsf{v}_2}) = \overline{\mathsf{v}_2 = \mathsf{v}_2 \wedge \overline{\mathsf{v}} = \overline{\mathsf{v}}}$

These axioms suffice for proving Gödel's celebrated diagonalization lemma.

REMARK

Of course, there is no such cheap way to Gödel's theorems. Gödel showed that the functions sub and q (and further operations) can be defined in an arithmetical theory for numerical codes of expressions. To this end he proved that all recursive functions can be represented in a fixed arithmetical system. And then he proved that the operation of substitution etc. are recursive. This requires some work and ideas.

diagonalization

The diagonalization function dia is defined in the following way:

Definition

$$\mathsf{dia}(x) = \mathsf{sub}(x, \bar{\mathsf{v}}, \mathsf{q}(x))$$

Remark

There are at least two ways to understand the syntactical status of dia. It may be considered an additional unary functionof $\mathcal{L}$, and the above equation is then an additional axiom of E. Alternatively, one can conceive dia as a metalinguistic abbreviation, which does not form part of the language $\mathcal{L}$, but which is just short notation for a more complex expression. This situation will encountered in the following frequently.

The diagonalization function dia is defined in the following way:

Definition

$$\mathsf{dia}(x) = \mathsf{sub}(x, \bar{\mathsf{v}}, \mathsf{q}(x))$$

remark

There are at least two ways to understand the syntactical status of dia. It may be considered an additional unary functionof $\mathcal{L}$, and the above equation is then an additional axiom of E. Alternatively, one can conceive dia as a metalinguistic abbreviation, which does not form part of the language $\mathcal{L}$, but which is just short notation for a more complex expression. This situation will encountered in the following frequently.

Lemma

Assume $\varphi(\mathsf{v})$ is a formula not containing bound occurrences of v. *Then the following holds:*

$$\mathsf{E} \vdash \mathsf{dia}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))}) = \overline{\varphi(\mathsf{dia}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))}))}$$

Proof.

In E the following equations can be proved::

$$\begin{aligned}
\mathsf{dia}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))}) &= \mathsf{sub}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))}, \overline{\mathsf{v}}, \mathsf{q}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))})) \\
&= \mathsf{sub}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))}, \overline{\mathsf{v}}, \overline{\overline{\varphi(\mathsf{dia}(\mathsf{v}))}}) \\
&= \overline{\varphi(\mathsf{dia}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))}))}
\end{aligned}$$

□

Lemma

Assume $\varphi(\mathsf{v})$ is a formula not containing bound occurrences of v. Then the following holds:

$$\mathsf{E} \vdash \mathsf{dia}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))}) = \overline{\varphi(\mathsf{dia}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))}))}$$

Proof.

In E the following equations can be proved::

$$\begin{aligned}
\mathsf{dia}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))}) &= \mathsf{sub}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))}, \bar{\mathsf{v}}, \mathsf{q}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))})) \\
&= \mathsf{sub}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))}, \bar{\mathsf{v}}, \overline{\overline{\varphi(\mathsf{dia}(\mathsf{v}))}}) \\
&= \overline{\varphi(\mathsf{dia}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))}))}
\end{aligned}$$

□

Theorem (diagonalization)

If $\varphi(\mathsf{v})$ is a formula of $\mathcal{L}$ with no bound occurrences of v, then one can find a formula γ such that the following holds:

$$\mathsf{E} \vdash \gamma \leftrightarrow \varphi(\overline{\gamma})$$

Proof.

Choose as γ the formula $\varphi(\mathsf{dia}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))}))$. Then one has by the previous Lemma:

$$\mathsf{E} \vdash \underbrace{\varphi(\mathsf{dia}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))}))}_{\gamma} \leftrightarrow \varphi(\overline{\underbrace{\varphi(\mathsf{dia}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))}))}_{\gamma}})$$

$\square$

Theorem (diagonalization)

If $\varphi(\mathsf{v})$ is a formula of $\mathcal{L}$ with no bound occurrences of v, then one can find a formula γ such that the following holds:

$$\mathsf{E} \vdash \gamma \leftrightarrow \varphi(\overline{\gamma})$$

Proof.

Choose as γ the formula $\varphi(\mathsf{dia}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))}))$. Then one has by the previous Lemma:

$$\mathsf{E} \vdash \underbrace{\varphi(\mathsf{dia}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))}))}_{\gamma} \leftrightarrow \varphi(\overline{\underbrace{\varphi(\mathsf{dia}(\overline{\varphi(\mathsf{dia}(\mathsf{v}))}))}_{\gamma}})$$

$\square$

The diagonal lemma may be provable without strong diagonalization.

Tarski obtained the diagonal lemma with concatenation only.

The language of Peano arithmetic lacks function symbols for sub and q and thus a functional expression for dia.

Whether we have weak or strong diagonalization can make a difference (Heck 2015, Schindler 2014).

At this point I could say more about self-reference.

The diagonal lemma may be provable without strong diagonalization.

Tarski obtained the diagonal lemma with concatenation only.

The language of Peano arithmetic lacks function symbols for sub and q and thus a functional expression for dia.

Whether we have weak or strong diagonalization can make a difference (Heck 2015, Schindler 2014).

At this point I could say more about self-reference.

The diagonal lemma may be provable without strong diagonalization.

Tarski obtained the diagonal lemma with concatenation only.

The language of Peano arithmetic lacks function symbols for sub and q and thus a functional expression for dia.

Whether we have weak or strong diagonalization can make a difference (Heck 2015, Schindler 2014).

At this point I could say more about self-reference.

The diagonal lemma may be provable without strong diagonalization.

Tarski obtained the diagonal lemma with concatenation only.

The language of Peano arithmetic lacks function symbols for sub and q and thus a functional expression for dia.

Whether we have weak or strong diagonalization can make a difference (Heck 2015, Schindler 2014).

At this point I could say more about self-reference.

bibliography

References

Richard G. Heck. Consistency and the theory of truth. *Review of Symbolic Logic*, 8:424–466, 2015.

Willard Van Orman Quine. *Philosophy of Logic*. Harvard University Press, Cambridge, Massachusetts, 1970.

Thomas Schindler. *Type-free truth*. PhD thesis, Ludwig-Maximilians-Universität München, 2014.