Meta-Logic: Soundness and Completeness

Robert Keller

October 2007

Meta-?

- "Meta-Logic" or "Meta-Mathematics" means proving things ***about*** logic rather than just ***within*** logic.
- For example, we might want to prove something about all proofs or all theorems, or that a certain formula is ***not*** a theorem.
- The language ***about*** which we prove things is called the **object** language.
- The language ***within*** which we prove things about the object language is the **meta-language**.

Validity vs. Provability

- The symbols $\vdash$ and $\models$ are part of the meta-language.
- $\varphi_1, \ldots, \varphi_n \vdash \psi$ means ψ is **provable** from $\varphi_1, \ldots, \varphi_n$ (sequent)
- $\varphi_1, \ldots, \varphi_n \models \psi$ means $\varphi_1, \ldots, \varphi_n$ **entail** ψ:

  If each of φ_i is true, then ψ is true. (entailment)
- $\vdash \psi$ and $\models \psi$ are the special case with $n = 0$.

Validity vs. Provability

- Generally, if Γ is (possibly-infinite) set of formulas
- The symbols $\vdash$ and $\models$ are part of the meta-language.
- $\Gamma \vdash \psi$ means ψ is **provable** from formulas Γ
- $\Gamma \models \psi$ means:

  Any valuation that satisfies Γ also satisfies ψ.

[A valuation **satisfies a formula** if it induces the value T.

A valuation **satisfies a set** of formulas if it satisfies each formula in the set.]

Satisfiability

- Γ is **satisfiable** if there is a valuation that satisfies it.
- **Lemma S**: Γ is **satisfiable** iff not $\Gamma \models \perp$.
- Proof: Suppose that Γ is **satisfiable**. Let ν be a valuation that satisfies it. ***No*** valuation satisfies $\perp$. So it is ***not*** the case that every valuation satisfying Γ satisfies $\perp$,
  i.e. "not $\Gamma \models \perp$".

  Conversely, suppose "not $\Gamma \models \perp$". This says there is ***some*** valuation which satisfies Γ but does ***not*** satisfy $\perp$. But all valuations don't satisfy $\perp$. So there is simply some valuation which satisfies Γ, i.e. Γ is satisfiable.

Soundness vs. Completeness of a logical system

- **Soundness**: Every provable sequent is an entailment:

  (for every Γ and ψ):

  $$\Gamma \vdash \psi \text{ implies } \Gamma \models \psi$$

- **Completeness**: Every valid sequent is provable:

  (for every Γ and ψ):

  $$\Gamma \models \psi \text{ implies } \Gamma \vdash \psi$$

Recall Definition of "Truth" for the Propositional Case

- A **valuation** is a mapping from proposition symbols $\{p, q, r, \ldots\}$ to the set $\{$**T**, **F**$\}$.

- A valuation ν is **extended** to an arbitrary formulas inductively as follows:
  - $\nu(\top)$ = **T**.
  - $\nu(\bot)$ = **F**.
  - $\nu(\varphi \wedge \psi)$ = **T** iff $\nu(\varphi)$ = **T** and $\nu(\psi)$ = **T**.
  - $\nu(\varphi \vee \psi)$ = **T** iff $\nu(\varphi)$ = **T** or $\nu(\psi)$ = **T**.
  - $\nu(\varphi \rightarrow \psi)$ = **T** iff $\nu(\varphi)$ = **F** or $\nu(\psi)$ = **T**.
  - $\nu(\neg \varphi)$ = **T** iff $\nu(\varphi)$ = **F**.

Definition of Entailment

- $\Gamma \models \psi$ means

  For each valuation ν:

  If, for each φ in Γ, $\nu(\varphi) = \mathbf{T}$

  then also $\nu(\psi) = \mathbf{T}$.

- When $\models \psi$ (i.e. Γ is empty), ψ is called a ***tautology.***

Proof of Soundness

- **Soundness**: Every sequent of Natural Deduction is an entailment:

  (for every Γ, ψ):

  $\Gamma \vdash \psi$ implies $\Gamma \models \psi$

  and $\vdash \psi$ implies $\models \psi$

- Assume that $\Gamma \vdash \psi$, to show $\Gamma \models \psi$.
- This will be by **structural induction** on the proof of ψ from formulas in Γ.

Proof Trees

- Recall that every ND proof can be expressed as a **tree**.
- The root of the tree is the conclusion ψ of the proof.
- The leaves of the tree are premises and uncancelled hypotheses: formulas in Γ.
- Nodes are rules.
- **Structural induction:**
- **Basis**: The simplest proof is a tree of one node. That node is justified in one of two ways:
  - The consequent of a rule with no antecedent.
  - A premise.
- There are none of the first kind of rule in propositional calculus.
- In the second case, a premise must be in Γ:
  - If for each φ in Γ, $\nu(\varphi)$ = **T** then also $\nu(\psi)$ = **T**, since ψ in Γ.
  - Hence $\Gamma \models \psi$.

Proof of Soundness Continued

- **Induction Step:**
- A proof tree of >1 nodes has the conclusion as the root.
- The conclusion must have come from a rule application.
- The antecedents of the rule are roots of proof sub-trees.
- The leaves of the sub-trees must be formulas in Γ, or hypotheses cancelled by applying the final rule.
- We need to show that if ν satisfies all leaves of the first kind, then it satisfies the root as well.
- This is done by examining each rule in turn, to see that it "propagates satisfaction".

Truth Propagation for $(\wedge I)$

- $$\frac{\varphi \quad \psi}{\varphi \wedge \psi}$$

- φ and ψ are roots of their respective trees. Suppose ν satisfies the leaves of both of those trees. Then by the inductive hypothesis, it satisfies φ and ψ.

- By the truth table definition for ν, must satisfy $\varphi \wedge \psi$.

Truth Propagation for ($\wedge$E)

- $$\frac{\varphi \wedge \psi}{\varphi} \quad (\wedge E)$$

- Suppose ν satisfies the leaves of the overall tree. Then it must satisfy the leaves of the sub-tree with $\varphi \wedge \psi$ as root.

- Thus from the truth table for $\wedge$, it must satisfy φ.

Truth Propagation for $(\vee I)$

- $$\frac{\varphi}{\varphi \vee \psi} \quad (\vee I)$$

- Suppose v satisfies the leaves of the overall tree. Then it must satisfy the leaves of the sub-tree with φ as root.

- Thus, from the truth table for $\vee$, it must also satisfy $\varphi \vee \psi$.

Truth Propagation for $\rightarrow$-Elimination Rule

- $$\frac{\varphi, \quad \varphi\rightarrow\psi}{\psi} \qquad (\rightarrow E)$$

- Suppose that ν satisfies the leaves of the tree with ψ as root. Then it also satisfies the sub-trees with each of φ and $\varphi\rightarrow\psi$ as roots.

- From the truth table for $\rightarrow$, ν must also satisfy ψ.

The Rules Employing Sub-Proofs

- Here's where the going gets a little tricky.
- Sub-proofs have hypotheses that get cancelled.

Truth Propagation for $\rightarrow I$

$$\dfrac{\boxed{\begin{array}{c}\varphi \\ \vdots \\ \psi\end{array}}}{\varphi\rightarrow\psi}\ (\rightarrow I)$$

- Suppose ν is a valuation that satisfies the leaves of the tree with $\varphi \rightarrow\psi$ as root.
- The sub-proof of ψ from φ will generally use formulas derived from those leaves, and from φ, which is not considered a leaf overall, since it gets cancelled.
- However, the proof tree for ψ does have φ for a leaf.
- If ν doesn't satisfy φ, then it does satisfy $\varphi\rightarrow\psi$.
- If ν satisfies φ, then from the inductive assumption, it satisfies the leaf ψ. Hence it satisfies $\varphi\rightarrow\psi$ in this case as well.

Truth Propagation for ∨E

- This rule uses two sub-derivations:

$$\dfrac{\varphi \vee \psi \qquad \boxed{\begin{array}{c}\varphi\\ .\\ .\\ \chi\end{array}} \quad \boxed{\begin{array}{c}\psi\\ .\\ .\\ \chi\end{array}}}{\chi}\ (\vee E)$$

- The argument is similar to that for $\rightarrow$I, but now we consider a ν that satisfies the leaves of the overall proof. $\varphi \vee \psi$ is such a leaf, so by the truth table for $\vee$, one of φ or ψ must be satisfied by ν. If ν happens to satisfy φ, then by the induction hypothesis, ν satisfies χ. Similarly if ν satisfies ψ. Thus in either case ν satisfies χ.

Truth Preservation for $\neg$I

$$\dfrac{\boxed{\begin{array}{c}\varphi\\ \vdots\\ \bot\end{array}}}{\neg\varphi}\ (\neg i)$$

This is like $\rightarrow$I, since $\neg\varphi$ can be treated as an abbreviation for $\varphi\rightarrow\bot$.

Truth Preservation for ¬E

$$\frac{\varphi \wedge \neg\varphi}{\bot} \qquad (\neg E)$$

This is like $\rightarrow$E, since $\neg\varphi$ can be treated as an abbreviation for $\varphi\rightarrow\bot$.

Truth Preservation for $\perp$E

$$\frac{\perp}{\varphi} \quad (\perp E)$$

No valuation ν satisfies $\perp$.

Thus for every valuation ν that does, ν satisfies φ as well.

Truth Preservation for RAA

$$\dfrac{\boxed{\begin{array}{c}\neg\varphi \\ \vdots \\ \bot\end{array}}}{\varphi}\ (\mathrm{RAA})$$

Consider a valuation ν that satisfies the leaves of the overall proof. If ν happens to satisfy φ we are done. If ν does not satisfy φ, then it satisfies $\neg\varphi$.

By the induction hypothesis, ν satisfies $\bot$. But this is impossible. So in all cases, ν satisfies $\neg\varphi$.

Completion of Soundness Proof

- Having addressed all of the ND rules, the proof of soundness for propositional natural deduction is complete, by structural induction.

Uses of Soundness

- There is an algorithm for determining whether or not

$$\varphi_1, \ldots, \varphi_n \models \psi$$

- Thus, one can compute a **necessary** condition of whether there is a proof of

$$\varphi_1, \ldots, \varphi_n \vdash \psi$$

- In other words, before embarking on trying to find a proof of a formula, we can sometimes check whether the formula follows on semantic grounds first.

Completeness

- Completeness says (for all Γ, ψ)

$$\Gamma \models \psi \text{ implies } \Gamma \vdash \psi$$

- The general case will require a "non-constructive" proof, since Γ could be infinite.
- The case of Γ finite is special, and admits a constructive proof.

Finite Completeness

- Finite completeness says (for all $\varphi_1, \ldots, \varphi_n, \psi$)

  $$\varphi_1, \ldots, \varphi_n \models \psi$$

  implies

  $$\varphi_1, \ldots, \varphi_n \vdash \psi$$

- If this could be established, then the algorithm mentioned for soundness would be a necessary and **sufficient** condition for the existence of a proof. That is, provability would be **solvability**.

Proof of Finite Completeness

Three steps are used:

1. $\varphi_1, \ldots, \varphi_n \models \psi$ implies $\models (\varphi_1 \rightarrow (\varphi_2 \rightarrow \ldots (\varphi_n \rightarrow \psi)) \ldots)$

2. For any formula η, $\models \eta$ implies $\vdash \eta$.

3. $\vdash (\varphi_1 \rightarrow (\varphi_2 \rightarrow \ldots (\varphi_n \rightarrow \psi)) \ldots)$ implies $\varphi_1, \ldots, \varphi_n \vdash \psi$

   Step 2 is the key one, as only it bridges the gap between $\models$ and $\vdash$. The other two are simplifying steps.

Proof that $\models (\varphi_1 \rightarrow (\varphi_2 \rightarrow \ldots (\varphi_n \rightarrow \psi)) \ldots)$

implies $\vdash (\varphi_1 \rightarrow (\varphi_2 \rightarrow \ldots (\varphi_n \rightarrow \psi)) \ldots)$

- Given a formula of the indicated form that is true for every valuation, we need to construct a proof.

- The proof that would be constructed by the uniform method that will be developed might not be the one that we'd give left on our own. It will generally be more complex than necessary.

Proof that for all η
$\models \eta$ implies $\vdash \eta$

- Assume $\models \eta$.
- Let $p_1, p_2, \ldots, p_k$ be the set of all proposition symbols that occur in η.
- For each combination of proposition symbols with and without negation, show that there is a sequent with that combination on the left and the formula of interest on the right:
  - $p_1, \ p_2, \ldots, p_k \vdash \eta$
  - $\neg p_1, \ p_2, \ldots, p_k \vdash \eta$
  - $p_1, \neg p_2, \ldots, p_k \vdash \eta$
  - $\neg p_1, \neg p_2, \ldots, p_k \vdash \eta$ etc.
- Then those 2^k sequents will be combined into a single sequent of the required form.

The Combination Process

- Because this constructs a derivation that is of length exponential in k, we will show it by example, for k = 2.

- Given that we have:
  - $p_1, \ p_2 \vdash \eta$
  - $\neg p_1, \ p_2 \vdash \eta$
  - $p_1, \neg p_2 \vdash \eta$
  - $\neg p_1, \neg p_2 \vdash \eta$

- The proof constructed for the single sequent is shown on the next page.

Proof Constructed for the Single Sequent

1.	$p_1 \vee \neg p_1$	LEM (a derived rule)
2.	p_1	Assumption
3.	$p_2 \vee \neg p_2$	LEM
4.	p_2	Assumption
	. . . steps in the proof of $\mathbf{p_1, p_2} \vdash \eta$	
5.	η	
6.	$\neg p_2$	Assumption
	. . . steps in the proof of $\mathbf{p_1, \neg p_2} \vdash \eta$	
7.	η	
8.	η	$\vee$E 3, 4-5, 6-7
9.	$\neg p_1$	Assumption
10.	$p_2 \vee \neg p_2$	LEM
11.	p_2	Assumption
	. . . steps in the proof of $\mathbf{\neg p_1, p_2} \vdash \eta$	
12.	η	
13.	$\neg p_2$	Assumption
	. . . steps in the proof of $\mathbf{\neg p_1, \neg p_2} \vdash \eta$	
14.	η	
15.	η	$\vee$E 10, 11-12, 13-14
16.	η	$\vee$E 1, 2-8, 9-15

Proofs for the Individual Sequents

- We are left with showing that each of the individual sequents
  - $p_1, \; p_2, \ldots, p_k \vdash \eta$
  - $\neg p_1, \; p_2, \ldots, p_k \vdash \eta$
  - $p_1, \neg p_2, \ldots, p_k \vdash \eta$
  - $\neg p_1, \neg p_2, \ldots, p_k \vdash \eta$ etc.

  has a proof, given that
  - $\models \eta$.

Proofs for the Individual Sequents

- For any formula η, we want to show that $\models \eta$ implies each of the individual sequents below has a proof
  - $p_1, \; p_2, \ldots, p_k \vdash \eta$
  - $\neg p_1, \; p_2, \ldots, p_k \vdash \eta$
  - $p_1, \neg p_2, \ldots, p_k \vdash \eta$
  - $\neg p_1, \neg p_2, \ldots, p_k \vdash \eta$ etc.

  where $p_1, p_2, \ldots, p_k$ are the proposition symbols in η.

- Consider any combination $p^*_1, p^*_2, \ldots, p^*_k$ of the symbols negated or un-negated (e.g. $\neg p_1, p_2, \ldots, \neg p_k$) and the corresponding **valuation** that makes $\nu(p^*_1 \wedge p^*_2 \wedge \ldots \wedge p^*_k) = \mathbf{T}$.

- **Lemma:**

  A: If $\nu(\eta) = \mathbf{T}$ then $p^*_1, p^*_2, \ldots, p^*_k \vdash \eta$.

  B: If $\nu(\eta) = \mathbf{F}$ then $p^*_1, p^*_2, \ldots, p^*_k \vdash (\neg\eta)$.

Proving A. If $\nu(\eta)$ = **T** then $p^*_1, p^*_2, \ldots, p^*_k \vdash \eta$.

B. If $\nu(\eta)$ = **F** then $p^*_1, p^*_2, \ldots, p^*_k \vdash (\neg\eta)$.

- This is done by **structural induction** on the **structure** of the **formula** η.
- **Basis**: If η is a single proposition symbol p, then
  - If $\nu(p)$ = **T**, then p^* must be p, and we certainly have $p \vdash p$ (so A).
  - If $\nu(p)$ = **F** , then p^* must be $\neg p$, and we have $\neg p \vdash \neg p$ (so B).
  - If η is $\perp$, then $\nu(\perp)$ = **F** always, but also $\vdash \neg\perp$ (by $\neg$I)(so B).
- Induction Step: We have to show that the inductive hypothesis implies the conclusion for each possible operator: $\neg \wedge \vee \rightarrow$.

Case where η is of form $\neg\rho$ for some ρ :

- If $\nu(\eta)$ = **T**, then $\nu(\rho)$ = **F**. By the induction hypothesis, part 2:

  $$p^*_1, p^*_2, \ldots, p^*_k \vdash (\neg\rho),$$

  i.e.

  $$p^*_1, p^*_2, \ldots, p^*_k \vdash \eta, \text{ (case A).}$$

- If $\nu(\eta)$ = **F**, then $\nu(\rho)$ = **T**. By the induction hypothesis, part 1:

  $$p^*_1, p^*_2, \ldots, p^*_k \vdash \rho.$$

  Using $\neg\neg$I to extend the proof one step, we have

  $$p^*_1, p^*_2, \ldots, p^*_k \vdash \neg(\neg\rho).$$

  Therefore

  $$p^*_1, p^*_2, \ldots, p^*_k \vdash \neg\eta, \text{ (case B).}$$

Case where η is of form $\rho_1 \wedge \rho_2$

- We need to consider 4 cases:
  $\nu(\rho_1, \rho_2)$ = **FF**, **FT**, **TF**, and **TT**.
- **FF**:
  By the induction hypothesis
  $p^*_1, p^*_2, \ldots, p^*_k \vdash (\neg \rho_1)$
  $p^*_1, p^*_2, \ldots, p^*_k \vdash (\neg \rho_2)$.
  Using $\neg$I, we get, in several steps, a proof of
  $p^*_1, p^*_2, \ldots, p^*_k \vdash \neg(\rho_1 \wedge \rho_2)$
  This conforms to the fact that $\nu(\rho_1 \wedge \rho_2)$ = **F** (case B).
- **The other 3 cases for $\rho_1 \wedge \rho_2$ are similar.**
- **The cases for the other operators ($\vee$, $\rightarrow$) are similar.**
- This concludes our sketch of the proof of the propositional completeness theorem.

Algorithm-Based Proof

- The proof just outlined is sufficiently constructive that we can create an algorithm from it:

- Given a tautology η, generate a natural deduction proof of η.

- In some sense, such an algorithmic proof is useful, in that it can be actively tested for examples, unlike an ordinary proof.

Completeness in the General Propositional Case: Consistency

- **Definition**: A set of formulas Γ is **consistent** provided

  not $\Gamma \vdash \perp$.

- Note the parallel:
  - **Consistency** of Γ: Not $\Gamma \vdash \perp$.
  - **Satisfiability** of Γ: Not $\Gamma \models \perp$.

Completeness in the General Propositional Case

- **Completeness Stmt I**: For all Γ, φ: $\Gamma \models \varphi$ implies $\Gamma \vdash \varphi$.

- **Completeness Stmt II** (Contrapositive, thus equivalent to I):
  - For all Γ, φ: (not $\Gamma \vdash \varphi$) implies (not $\Gamma \models \varphi$).

- Consistency of Γ: Not $\Gamma \vdash \perp$.
- Satisfiability of Γ: Not $\Gamma \models \perp$. (from Lemma S)

- **Completeness Stmt III**:
  - For all Γ: (not $\Gamma \vdash \perp$) implies (not $\Gamma \models \perp$),
    i.e.

Γ consistent implies Γ satisfiable.

Justification

- **Completeness II implies Completeness III** :
  - Simply take φ to be $\bot$.
- **Completeness III implies Completeness I** :
  - Suppose III: For all Γ: (not $\Gamma \vdash \bot$) implies (not $\Gamma \models \bot$).
  - The contrapositive is: For all Γ: $\Gamma \models \bot$ implies $\Gamma \vdash \bot$ (*).
  - To get I, assume $\Gamma \models \varphi$, and show $\Gamma \vdash \varphi$:
  - From $\Gamma \models \varphi$, $\Gamma \cup \{\neg\varphi\} \models \bot$ (any valuation satisfying Γ satisfies φ, so there can be no valuation satisfying $\neg\varphi$).
  - From (*), $\Gamma \cup \{\neg\varphi\} \vdash \bot$.
  - Using the RAA rule then, $\Gamma \vdash \varphi$.

General Completeness Theorem

- We have shown that completeness is equivalent to:
- (For all Γ)
  Γ consistent implies Γ satisfiable.
- Sketch:
  To prove this, we start with a Γ_0 that is consistent, to eventually show there is a valuation satisfying Γ_0.

Sketch, continued

- First we extend Γ_0 to a maximally consistent set Γ_{max}:
  - Let Γ be Γ_0.
  - Enumerate every possible formula φ.
    - If $\Gamma \cup \{\varphi\}$ is consistent, add φ to Γ.
  - The Γ in the limit of this process is Γ_{max}.

- Then show that Γ_{max} is consistent, and in fact, maximally consistent.

Sketch, continued

- Γ_{max} is consistent, because at no step did we add a formula that would destroy its consistency.
- It is maximally consistent because it is closed under derivability: If $\Gamma_{max} \vdash \varphi$, then in fact $\varphi \in \Gamma_{max}$.
- We then show that any maximally consistent set has a valuation satisfying it. **Define** a valuation ν as follows:
  - For each proposition symbol p, if $p \in \Gamma_{max}$ then $\nu(p) = T$, otherwise $\nu(p) = F$.
- Then argue that ν satisfies Γ_{max} using closure under derivability.
- Finally ν also satisfies Γ_0, since $\Gamma_0 \subseteq \Gamma_{max}$.

Completeness in the General Predicate Case

- **Just as in the Propositional Case:**

  A set of formulas Γ is **consistent** provided

  $$\text{not } \Gamma \vdash \bot.$$

- Note the parallel:
  - **Consistency** of Γ: Not $\Gamma \vdash \bot$.
  - **Satisfiability** of Γ: Not $\Gamma \models \bot$.
- Another way of saying the latter is "Γ has a model."

Completeness Theorem

- In the form usually proved :

  If a set of formulas Γ is consistent,
  then Γ has a model.

which, as we showed previously, is equivalent to:

For all φ: $\Gamma \models \varphi$ implies $\Gamma \vdash \varphi$

Proof of the Completeness Theorem

- Parallels the propositional case in many ways:
  - Construct a maximally-consistent set from the original consistent set.
  - Construct a model for the maximally-consistent set.

How to construct a model out of nothing but formulas?

- Define a universe (called the Herbrand Universe), the members of which are terms.
- Example: If there is one constant symbol c and one binary function symbol f, then the Herbrand universe is

  {c, f(c, c), f(c, f(c, c)), f(f(c, c), c), … }

Model construction, continued

- Create an interpretation with the universe as its universe:
  - The function symbols are interpreted in the "obvious" way.
  - The predicate symbols are interpreted so as to agree with atomic formulas in the maximally-consistent set.
  - New constants (called Henkin-constants) must be introduced into the language to provide constants that solve ∃-formulas. However, these constants do not change the derivability of expressions in the original language.
  - It is shown that this interpretation is a model for the original formulas.

Theories vs. Frameworks

- The natural deduction **framework** is both sound and complete.
- However, there are **theories** that are incomplete.
- This is the source of some confusion.

Theory

- A **theory** is a set of formulas closed under derivability. The formulas are called **theorems**.

- Usually the set is based on a smaller set of **axioms**.

- The set of axioms may be:
  - Finite
  - Computable
  - or neither, but the first two are the most useful.

Examples of a Theory

- Peano arithmetic (PA)
- Theory of groups
- Set theory (e.g. ZFC)

Completeness of a theory

- A theory is complete if, for any closed formula, either:
  - The formula is provable, or
  - The negation of the formula is provable.

Gödel’s Incompleteness Theorem

- (paraphrased)

  Peano Arithmetic is Incomplete:

- There is a formula φ such that neither φ nor $\neg\varphi$ is provable.

Gödel’s Formula

- “This formula is not provable.”

- More precisely: Construct $\varphi(n)$ such that:
  $\varphi(n)$ means the formula whose number is designated by the term n is *not* provable.

- Every PA formula has such a term T, including $\varphi(n_\varphi)$ where n_φ is the term for $\varphi(n_\varphi)$.

- Hence, assuming PA is consistent, $\varphi(n_\varphi)$ could not be provable.

Gödel's Proof

- More precisely: Construct $\varphi(n)$ such that:
  $\varphi(n)$ means the formula whose number is designated by the term n is *not* provable.
- Every PA formula has such a term T, including $\varphi(n_\varphi)$ where n_φ is the term for $\varphi(n_\varphi)$.
- If $\varphi(n_\varphi)$ were provable, then (by consistency) it would be *true* in PA.
- But if $\varphi(n_\varphi)$ is true, then it could not be provable.
- And if $\neg\varphi(n_\varphi)$ were provable, then it would be false in PA, so $\varphi(n_\varphi)$ would be provable.

Who Cares?

- Hilbert, …

Decidable theories

- A theory is **decidable** if there is an algorithm that will tell whether or not any given formula is a theorem.

- Complete (+ recursively axiomatized) $\Rightarrow$ Decidable