Classical Logic

First published Sat Sep 16, 2000; substantive revision Mon Nov 2, 2009

Typically, a *logic* consists of a formal or informal language together with a deductive system and/or a model-theoretic semantics. The language is, or corresponds to, a part of a natural language like English or Greek. The deductive system is to capture, codify, or simply record which *inferences* are correct for the given language, and the semantics is to capture, codify, or record the meanings, or truth-conditions, or possible truth conditions, for at least part of the language.

The following sections provide the basics of a typical logic, sometimes called "classical elementary logic" or "classical first-order logic". Section 2 develops a formal language, with a rigorous syntax and grammar. The formal language is a recursively defined collection of strings on a fixed alphabet. As such, it has no meaning, or perhaps better, the meaning of the formulas is given by the deductive system and the semantics. Some of the symbols have counterparts in ordinary language. We define an *argument* to be a non-empty collection of formulas in the formal language, one of which is designated to be the conclusion. The other formulas (if any) in an argument are its premises. Section 3 sets up a deductive system for the language, in the spirit of natural deduction. An argument is *derivable* if there is a deduction from some of its premises to its conclusion. Section 4 provides a model-theoretic semantics. An argument is *valid* if there is no interpretation (in the semantics) in which its premises are all true and its conclusion false. This reflects the longstanding view that a valid argument is truth-preserving.

In Section 5, we turn to relationships between the deductive system and the semantics, and in particular, the relationship between derivability and validity. We show that an argument is derivable only if it is valid. This pleasant feature, called *soundness*, entails that no deduction takes one from true premises to a false conclusion. Thus, deductions preserve truth, and there aren't too many deductions. Then we establish a converse, called *completeness*, that an argument is valid only if it is derivable. This establishes that the deductive system is rich enough to provide a deduction for every valid argument. There are enough deductions. All and only valid arguments are derivable. We briefly indicate other features of the logic, some of which are corollaries to soundness and completeness.

---

1. Introduction

Today, logic is both a branch of mathematics and a branch of philosophy. In most large universities, both departments offer sequences of courses in logic, and there is usually a lot of overlap between them. Formal languages, deductive systems, and model-theoretic semantics are mathematical objects and, as such, the logician is interested in their mathematical properties and relations. Soundness, completeness, and most of the other results reported below are typical examples. Philosophically, logic is the study of *correct reasoning*. Reasoning is an epistemic, mental activity. This raises questions concerning the philosophical relevance of the mathematical aspects of logic. How do deducibility and validity, as properties of formal languages--sets of strings on a fixed alphabet--relate to correct reasoning? What do the mathematical results reported below have to do with the original philosophical issue? This is an instance of the philosophical problem of explaining how mathematics applies to non-mathematical reality.

Typically, ordinary reasoning takes place in a natural language, or perhaps a natural language augmented with some mathematical symbols. So our question begins with the relationship between a natural language and a formal language. Without attempting to be comprehensive, it may help to sketch several options on this matter.

One view is that the formal languages accurately exhibit actual features of certain fragments of a natural language. Some philosophers claim that declarative sentences of natural language have underlying *logical forms* and that these forms are displayed by formulas of a formal language. Other writers hold

that (successful) declarative sentences express *propositions*; and formulas of formal languages somehow display the forms of these propositions. On views like this, the components of a logic provide the underlying deep structure of correct reasoning. A chunk of reasoning in natural language is correct if the forms underlying the sentences constitute a valid or deducible argument. See for example, Montague [1974], Davidson [1984], Lycan [1984].

Another view, held at least in part by Gottlob Frege and Wilhelm Leibniz, is that because natural languages are vague and ambiguous, they should be *replaced* by formal languages. A similar view, held by W. V. O. Quine (e.g., [1960], [1986]), is that a natural language should be *regimented*, cleaned up for serious scientific and metaphysical work. One desideratum of the enterprise is that the logical structures in the regimented language should be transparent. It should be easy to "read off" the logical properties of each sentence. A regimented language is similar to a formal language regarding, for example, the explicitly presented rigor of its syntax and its truth conditions.

On a view like this, deducibility and validity represent *idealizations* of correct reasoning in natural language. A chunk of reasoning is correct to the extent that it corresponds to, or can be regimented by, a valid or deducible argument in a formal language.

When mathematicians and many philosophers reason, they occasionally invoke formulas in a formal language to help disambiguate, or otherwise clarify what they mean. In other words, sometimes formulas in a formal language are *used* in ordinary reasoning. This suggests that one might think of a formal language as an *addendum* to a natural language. Then our present question concerns the relationship between this addendum and the original language. What do deducibility and validity, as sharply defined on the addendum, tell us about correct reasoning in general?

Another view is that a formal language is a *mathematical model* of a natural language in roughly the same sense as, say, a collection of point masses is a model of a system of physical objects, and the Bohr construction is a model of an atom. In other words, a formal language displays certain features of natural languages, or idealizations thereof, while ignoring or simplifying other features. The purpose of mathematical models is to shed light on what they are models of, without claiming that the model is accurate in all respects or that the model

should replace what it is a model of. On a view like this, deducibility and validity represent mathematical models of (perhaps different aspects of) correct reasoning in natural languages. Correct chunks of reasoning correspond, more or less, to valid or deducible arguments; incorrect chunks of reasoning roughly correspond to invalid or non-deducible arguments. See, for example, Corcoran [1973] or Shapiro [1998].

There is no need to adjudicate this matter here. Perhaps the truth lies in a combination of the above options, or maybe some other option is the correct, or most illuminating one. I raise the matter only to lend some philosophical perspective to the formal treatment that follows.

2. Language

Here we develop the basics of a formal language, or to be precise, a class of formal languages. Again, a formal language is a recursively defined set of strings on a fixed alphabet. Some aspects of the formal languages correspond to, or have counterparts in, natural languages like English. Technically, this "counterpart relation" is not part of the formal development, but I will mention it from time to time, to motivate some of the features and results.

Building blocks

We begin with analogues of *singular terms*, linguistic items whose function is to denote a person or object. We call these *terms*. We assume a stock of *individual constants*. These are lower-case letters, near the beginning of the Roman alphabet, with or without numerical subscripts:

$a, a_1, b_{23}, c, d_{22}$, etc.

We envisage a potential infinity of individual constants. In the present system each constant is a single character, and so individual constants do not have an internal syntax. Thus we have an infinite alphabet. This last could be avoided by taking a constant like d_{22}, for example, to consist of three characters, a lowercase "d" followed by a pair of subscript "2"s.

We also assume a stock of *individual variables*. These are lower-case letters, near the end of the alphabet, with or without numerical subscripts:

w, *x*, y_{12}, *z*, z_4, etc.

Variables serve a dual function. Sometimes a variable is used as a singular term to denote a specific, but unspecified (or arbitrary) object. For example, a mathematician might start a derivation: "Let *x* be a natural number". Variables are also used to express generality, as in the mathematical assertion that for any natural number *x*, there is a natural number *y*, such that $y>x$ and *y* is prime. Some logicians employ different symbols for unspecified objects (sometimes called "individual parameters") and variables used to express generality.

Constants and variables are the only terms in our formal language, so all of our terms are simple, corresponding to proper names and pronouns. Some authors also introduce *function letters*, which allow complex terms corresponding to: "7+4" and "the wife of Bill Clinton", or complex terms containing variables, like "the father of *x*" and "*x*/*y*". Logic books aimed at mathematicians are likely to contain function letters, probably due to the centrality of functions to mathematical discourse. Books aimed at a more general audience (or at philosophy students), may leave out function letters, since it simplifies the syntax and theory. We follow the latter route here. This is an instance of a general tradeoff between presenting a system with greater expressive resources, at the cost of making its formal treatment more complex.

For each natural number *n*, we introduce a stock of *n*-place*predicate letters*. These are upper-case letters at the beginning or middle of the alphabet. A superscript indicates the number of places, and there may or may not be a subscript. For example,

A^3, B^3_2, P^3, etc.

are three-place predicate letters. We often omit the superscript, when no confusion will result. We also add a special two-place symbol "=" for identity.

Zero-place predicate letters are sometimes called "sentence letters". They correspond to free-standing sentences whose internal structure does not matter. One-place predicate letters, called "monadic predicate letters", correspond to linguistic items denoting properties, like "being a man", "being red", or "being a prime number". Two-place predicate letters, "binary predicate letters", correspond to linguistic items denoting binary relations, like "is a parent of" or

"is greater than". Three-place predicate letters correspond to three-place relations, like "lies on a straight line between". And so on.

The *non-logical terminology* of the language consists of its individual constants and predicate letters. The symbol "=", for identity, is not a non-logical symbol. In taking identity to be logical, we provide explicit treatment for it in the deductive system and the model-theoretic semantics. Most authors do the same, but there is some controversy over the issue (Quine [1986, Chapter 5]). If K is a set of constants and predicate letters, then we give the fundamentals of a language L1K= built on this set of non-logical terminology. It may be called the *first-order language with identity* on K. A similar language that lacks the symbol for identity (or which takes identity to be non-logical) may be called L1K, the *first-order language without identity* on K.

Atomic formulas

If V is an n-place predicate letter in K, and t_1, …, t_n are terms of K (i.e., constants in K or variables), then $Vt_1 … t_n$ is an *atomic formula* of L1K=. Notice that the terms t_1, …, t_n need not be distinct. Examples of atomic formulas include:

P^4xaab, C^1x, C^1a, D^0, A^3abc.

The last one is an analogue of a statement that a certain relation (A) holds between three objects (a, b, c). If t_1 and t_2 are terms, then $t_1=t_2$ is an atomic formula of L1K=. It corresponds to an assertion that t_1 is identical to t_2.

If an atomic formula has no variables, then it is called an *atomic sentence*. If it does have variables, it is called an *open formula*. In the above list of examples, the first and second are open; the rest are sentences.

Compound formulas

We now introduce the final items of the lexicon:

¬, &, ∨, →, ∀, ∃, (,)

We give a recursive definition of a *formula* of L1K=:

1. All atomic formulas of L1K= are formulas of L1K=.

2. If θ is a formula of L1K=, then so is ¬θ.

Asserting a sentence corresponding to ¬θ is tantamount to denying the sentence corresponding to θ. The symbol "¬" is called "negation", and is a unary connective.

3. If θ and ψ are formulas of L1K=, then so is (θ & ψ).

The ampersand "&" corresponds to the English "and" (when "and" is used to connect sentences). So (θ & ψ) can be read "θ and ψ". The formula (θ & ψ) is called the "conjunction" of θ and ψ.

4. If θ and ψ are formulas of L1K=, then so is (θ ∨ ψ).

The wedge "∨" corresponds to "either … or … or both", so (θ ∨ ψ) can be read "θ or ψ". The formula (θ ∨ ψ) is called the "disjunction" of θ and ψ.

5. If θ and ψ are formulas of L1K=, then so is (θ → ψ).

The arrow "→" corresponds to "if … then … ", so (θ → ψ) can be read "if θ then ψ" or "θ only if ψ".

The symbols "&", "∨", and "→" are called "binary connectives", since they serve to "connect" two sentences into one. Some authors introduce (θ ↔ ψ) as an abbreviation of ((θ → ψ) & (ψ → θ)). The symbol "↔" is an analogue of the locution "if and only if".

6. If θ is a formula of L1K= and v is a variable, then ∀v θ is a formula of L1K=.

The symbol "∀" is called a *universal quantifier*, and is an analogue of "for all"; so ∀vθ can be read "for all v, θ".

7. If θ is a formula of L1K= and v is a variable, then ∃v θ is a formula of L1K=.

The symbol "∃" is called an *existential quantifier*, and is an analogue of "there exists" or "there is"; so ∃v θ can be read "there is a v such that θ".

8. That's all folks. That is, all formulas are constructed in accordance with rules (1)-(7).

Clause (8) allows us to do inductions on the complexity of formulas. If a certain property holds of the atomic formulas and is closed under the operations presented in clauses (2)-(7), then the property holds of all formulas.

We next define the notion of an occurrence of a variable being *free* or *bound* in a formula. All variables that occur in an atomic formula are free. If a variable occurs free (or bound) in θ or in ψ, then that same occurrence is free (or bound) in $\neg\theta$, $(\theta \& \psi)$, $(\theta \lor \psi)$, and $(\theta \rightarrow \psi)$. That is, the (unary and binary) connectives do not change the status of variables that occur in them. All occurrences of the variable v in θ are bound in $\forall v\, \theta$ and $\exists v\, \theta$. Any *free* occurrences of v in θ are bound by the initial quantifier. All other variables that occur in θ are free or bound in $\forall v\, \theta$ and $\exists v\, \theta$, as they are in θ. A variable that immediately follows a quantifier (as in "$\forall x$" and "$\exists y$") is neither free nor bound. We do not think of those as occurrences of the variable.

For example, in the formula $(\forall x(Axy \lor Bx)\ \& Bx)$, the occurrences of "$x$" in Axy and in the first Bx are bound by the quantifier. The occurrence of "y" and last occurrence of "x" are free. In $\forall x(Ax \rightarrow \exists xBx)$, the "$x$" in Ax is bound by the initial universal quantifier, while the other occurrence of x is bound by the existential quantifier. The above syntax allows this "overlap" of bound variables, and it does not create an ambiguity, but we will avoid such formulas, as a matter of taste and clarity.

Free variables correspond to place-holders, while bound variables are used to express generality. If a formula has no free variables, then it is called a *sentence*. If a formula has free variables, it is called *open*.

Features of the syntax

Before turning to the deductive system and semantics, I mention a few features of the language, as developed so far. This helps draw the contrast between formal languages and natural languages like English.

We assume at the outset that all of the categories are disjoint. For example, no connective is also a quantifier or a variable, and the non-logical terms are not also parentheses or connectives. Also, the items within each category are distinct. For example, the sign for disjunction does not do double-duty as the negation symbol, and perhaps more significantly, no two-place predicate is also a one-place predicate.

Theorem 1. Every formula of L1K= has the same number of left and right parentheses. Moreover, each left parenthesis corresponds to a unique right parenthesis, which occurs to the right of the left parenthesis. Similarly, each right parenthesis corresponds to a unique left parenthesis, which occurs to the left of the given right parenthesis. If a parenthesis occurs between a matched pair of parentheses, then its mate also occurs within that matched pair. In other words, parentheses that occur within a matched pair are themselves matched.

Proof: By clause (8), every formula is built up from the atomic formulas using clauses (2)-(7). The atomic formulas have no parentheses (by the policy that the categories are disjoint). Parentheses are introduced only in clauses (3)-(5), and each time they are introduced as a matched set. So at any stage in the construction of a formula, the parentheses are paired off.

One difference between natural languages like English and formal languages like L1K= is that the latter are not supposed to have any ambiguities. Our policy that the different categories of symbols do not overlap, and that no symbol does double-duty helps avoid the kind of ambiguity, sometimes called "equivocation", that occurs when a single word has two meanings: "I'll meet you at the bank." But there are other kinds of ambiguity. Consider the English sentence:

John is married, and Mary is single, or Joe is crazy.

It can mean that John is married and either Mary is single or Joe is crazy, or else it can mean that either both John is married and Mary is single, or else Joe is crazy. An ambiguity like this, due to different ways to parse the same sentence, is sometimes called an "amphiboly". If our formal language did not have the parentheses in it, it would have amphibolies. For example, there would be a "formula" *A* & *B* ∨ *C*. Is this supposed to be ((*A* & *B*) ∨ *C*), or is it (*A* & (*B* ∨ *C*))? The parentheses resolve what would be an amphiboly.

Can we be sure that there are no other amphibolies in our language? That is, can we be sure that each formula of L1K= can be put together in only one way? Showing this is our next task.

Let us temporarily use the term "unary marker" for the negation symbol (¬) or a quantifier followed by a variable (e.g., ∀*x*, ∃*z*).

Lemma 2. Each formula consists of a string of zero or more unary markers followed by either an atomic formula or a formula produced using a binary connective, via one of clauses (3)-(5).

Proof: We proceed by induction on the complexity of the formula or, in other words, on the number of formation rules that are applied. The Lemma clearly holds for atomic formulas. Let n be a natural number, and suppose that the Lemma holds for any formula constructed from n or fewer instances of clauses (2)-(7). Let θ be a formula constructed from $n+1$ instances. The Lemma holds if the last clause used to construct θ was either (3), (4), or (5). If the last clause used to construct θ was (2), then θ is $\neg\psi$. Since ψ was constructed with n instances of the rule, the Lemma holds for ψ (by the induction hypothesis), and so it holds for θ. Similar reasoning shows the Lemma to hold for θ if the last clause was (6) or (7). By clause (8), this exhausts the cases, and so the Lemma holds for θ, by induction.

Lemma 3. If a formula θ contains a left parenthesis, then it ends with a right parenthesis, which matches the leftmost left parenthesis in θ.

Proof: Here we also proceed by induction on the number of instances of (2)-(7) used to construct the formula. Clearly, the Lemma holds for atomic formulas, since they have no parentheses. Suppose, then, that the Lemma holds for formulas constructed with n or fewer instances of (2)-(7), and let θ be constructed with $n+1$ instances. If the last clause applied was (3)-(5), then the Lemma holds since θ itself begins with a left parenthesis and ends with the matching right parenthesis. If the last clause applied was (2), then θ is $\neg\psi$, and the induction hypothesis applies to ψ. Similarly, if the last clause applied was (6) or (7), then θ consists of a quantifier, a variable, and a formula to which we can apply the induction hypothesis. It follows that the Lemma holds for θ.

Lemma 4. Each formula contains at least one atomic formula.

The proof proceeds by induction on the number of instances of (2)-(7) used to construct the formula, and we leave it as an exercise.

Theorem 5. Let α, β be nonempty sequences of characters on our alphabet, such that $\alpha\beta$ (i.e α followed by β) is a formula. Then α is *not* a formula.

Proof: By Theorem 1 and Lemma 3, if α contains a left parenthesis, then the right parenthesis that matches the leftmost left parenthesis in α β comes at the end of α β, and so the matching right parenthesis is in β. So, α has more left parentheses than right parentheses. By Theorem 1, α is not a formula. So now suppose that α does not contain any left parentheses. By Lemma 2, α β consists of a string of zero or more unary markers followed by either an atomic formula or a formula produced using a binary connective, via one of clauses (3)-(5). If the latter formula was produced via one of clauses (3)-(5), then it begins with a left parenthesis. Since α does not contain any parentheses, it must be a string of unary markers. But then α does not contain any atomic formulas, and so by Lemma 4, α is not a formula. The only case left is where $\alpha\beta$ consists of a string of unary markers followed by an atomic formula, either in the form $t_1=t_2$ or $Pt_1 \dots t_n$. Again, if α just consisted of unary markers, it would not be a formula, and so α must consist of the unary markers that start α β, followed by either t_1 by itself, $t_1=$ by itself, or the predicate letter P, and perhaps some (but not all) of the terms $t_1, \dots, t_n$. In the first two cases, α does not contain an atomic formula, by the policy that the categories do not overlap. Since P is an n-place predicate letter, by the policy that the predicate letters are distinct, P is not an m-place predicate letter for any $m \neq n$. So the part of α that consists of P followed by the terms is not an atomic formula. In all of these cases, then, α does not contain an atomic formula. By Lemma 4, α is not a formula.

We are finally in position to show that there is no amphiboly in our language.

Theorem 6. Let θ be any formula of L1K=. If θ is not atomic, then there is one and only one among (2)-(7) that was the last clause applied to construct θ. That is, θ could not be produced by two different clauses. Moreover, no formula produced by clauses (3)-(7) is atomic.

Proof: By Clause (8), either θ is atomic or it was produced by one of clauses (2)-(7). Thus, the first symbol in θ must be either a predicate letter, a term, a unary marker, or a left parenthesis. If the first symbol in θ is a predicate letter or term, then θ is atomic. In this case, θ was not produced by any of (2)-(7), since all such formulas begin with something other than a predicate letter or term. If the first symbol in θ is a negation sign "¬", then was θ produced by clause (2), and not by any other clause (since the other clauses produce formulas that begin with either a quantifier or a left parenthesis). Similarly, if θ begins with a universal quantifier, then it was produced by clause (6), and not by any other

clause, and if θ begins with an existential quantifier, then it was produced by clause (7), and not by any other clause. The only case left is where θ begins with a left parenthesis. In this case, it must have been produced by one of (3)-(5), and not by any other clause. We only need to rule out the possibility that θ was produced by more than one of (3)-(5). To take an example, suppose that θ was produced by (3) and (4). Then θ is $(\psi_1 \& \psi_2)$ and θ is also $(\psi_3 \vee \psi_4)$, where ψ_1, ψ_2, ψ_3, and ψ_4 are themselves formulas. That is, $(\psi_1 \& \psi_2)$ is the very same formula as $(\psi_3 \vee \psi_4)$. By Theorem 5, ψ_1 cannot be a proper part of ψ_3, nor can ψ_3 be a proper part of ψ_1. So ψ_1 must be the same formula as ψ_3. But then "&" must be the same symbol as "$\vee$", and this contradicts the policy that all of the symbols are different. So θ was not produced by both Clause (3) and Clause (4). Similar reasoning takes care of the other combinations.

This result is sometimes called "unique readability". It shows that each formula is produced from the atomic formulas via the various clauses in exactly one way. If θ was produced by clause (2), then its *main connective* is the initial "$\neg$". If θ was produced by clauses (3), (4), or (5), then its *main connective* is the introduced "&", "$\vee$", or "$\rightarrow$", respectively. If θ was produced by clauses (6) or (7), then its *main connective* is the initial quantifier. I apologize for the tedious details. I included them to indicate the level of precision and rigor for the syntax.

3. Deduction

We now introduce a *deductive system*, *D*, for our languages. As above, we define an *argument* to be a non-empty collection of formulas in the formal language, one of which is designated to be the *conclusion*. If there are any other formulas in the argument, they are its *premises*. By convention, we use "Γ", "Γ'", "Γ_1", etc, to range over sets of formulas, and we use the letters "ϕ", "ψ", "θ", uppercase or lowercase, with or without subscripts, to range over single formulas. We write "Γ, Γ'" for the union of Γ and Γ', and "Γ, ϕ" for the union of Γ with $\{\phi\}$.

We write an argument in the form $\langle \Gamma, \phi \rangle$, where Γ is the set of premises and ϕ is the conclusion. Remember that Γ may be empty. We write $\Gamma \vdash \phi$ to indicate that ϕ is deducible from Γ, or, in other words, that the argument $\langle \Gamma, \phi \rangle$ is deducible in *D*. We may write $\Gamma \vdash_D \phi$ to emphasize the deductive system *D*. We write $\vdash \phi$ or $\vdash_D \phi$ to indicate that ϕ can be deduced (in *D*) from the empty set of premises.

The rules in *D* are chosen to match logical relations concerning the English analogues of the logical terminology in the language. Again, we define the deducibility relation by recursion. We start with a rule of assumptions:

(As) If ϕ is a member of Γ, then Γ ⊢ ϕ.

We thus have that {ϕ} ⊢ ϕ; each premise follows from itself. We next present two clauses for each connective and quantifier. The clauses indicate how to "introduce" and "eliminate" formulas in which each symbol is the main connective.

First, recall that "&" is an analogue of the English connective "and". Intuitively, one can deduce a formula in the form (θ & ψ) if one has deduced θ and one has deduced ψ. Conversely, one can deduce θ from (θ & ψ) and one can deduce ψ from (θ & ψ):

(&I) If Γ₁ ⊢ θ and Γ₂ ⊢ ψ, then Γ₁, Γ₂ ⊢ (θ & ψ).

(&E) If Γ ⊢ (θ & ψ) then Γ ⊢ θ; and if Γ ⊢ (θ & ψ) then Γ ⊢ ψ.

The name "&I" stands for "&-introduction"; "&E" stands for "&-elimination".

Since, the symbol "∨" corresponds to the English "or", (θ ∨ ψ) should be deducible from θ, and (θ ∨ ψ) should also be deducible from ψ:

(∨**I**) If Γ ⊢ θ then Γ ⊢ (θ ∨ ψ); if Γ ⊢ ψ then Γ ⊢ (θ ∨ ψ).

The elimination rule is a bit more complicated. Suppose that "θ or ψ" is true. Suppose also that ϕ follows from θ and that ϕ follows from ψ. One can reason that if θ is true, then ϕis true. If instead ψ is true, we still have that ϕ is true. So either way, ϕ must be true.

(∨**E**) If Γ₁ ⊢ (θ ∨ ψ), Γ₂, θ ⊢ ϕ and Γ₃, ψ ⊢ ϕ, then Γ₁, Γ₂, Γ₃ ⊢ ϕ.

For the next clauses, recall that the symbol "→" is an analogue of the English "if … then … " construction. If one knows, or assumes (θ → ψ) and also knows, or assumes θ, then one can conclude ψ. Conversely, if one deduces ψ from an assumption θ, then one can conclude that (θ → ψ).

(→**I**) If Γ, θ ⊢ ψ, then Γ ⊢ (θ → ψ).

(→**E**) If $\Gamma_1 \vdash (\theta \rightarrow \psi)$ and $\Gamma_2 \vdash \theta$, then $\Gamma_1, \Gamma_2 \vdash \psi$.

Our next clauses are for the negation sign, "¬". The underlying idea is that a formula ψ is inconsistent with its negation ¬ψ. They cannot both be true. We call a pair of formulas ψ, ¬ψ *contradictory opposites*. If one can deduce such a pair from an assumption θ, then one can conclude that θ is false, or, in other words, one can conclude ¬θ.

(**¬I**) If $\Gamma_1, \theta \vdash \psi$ and $\Gamma_2, \theta \vdash$ ¬ψ, then $\Gamma_1, \Gamma_2 \vdash$ ¬θ.

There is some controversy over the other rule for the negation sign.

By (As), we have that $\{A, \neg A\} \vdash A$ and $\{A, \neg A\} \vdash \neg A$. So by ¬I we have that $\{A\} \vdash \neg\neg A$. However, we do not have the converse yet. Intuitively, ¬¬θ corresponds to "it is not the case that it is not the case that" . One might think that this last is equivalent to θ, and we have a rule to that effect:

(**DNE**) If $\Gamma \vdash \neg\neg\theta$, then $\Gamma \vdash \theta$.

The name DNE stands for "double-negation elimination". This inference is rejected by philosophers and mathematicians who do not hold that each meaningful sentence is either true or not true.*Intuitionistic logic* does not sanction the inference in question (see, for example Dummett [1977]), but, again, classical logic does.

To illustrate the parts of the deductive system D presented thus far, I show that $\vdash (A \vee \neg A)$:

(i) $\{\neg(A \vee \neg A), A\} \vdash \neg(A \vee \neg A)$, by (As)

(ii) $\{\neg(A \vee \neg A), A\} \vdash A$, by clause (As).

(iii) $\{\neg(A \vee \neg A), A\} \vdash (A \vee \neg A)$, by (∨I), from (ii).

(iv) $\{\neg(A \vee \neg A)\} \vdash \neg A$, by (¬I), from (i) and (iii).

(v) $\{\neg(A \vee \neg A), \neg A\} \vdash \neg(A \vee \neg A)$, by (As)

(vi) $\{\neg(A \vee \neg A), \neg A\} \vdash \neg A$, by (As)

(vii) $\{\neg(A \vee \neg A), \neg A\} \vdash (A \vee \neg A)$, by (∨I), from (vi).

(viii) $\{\neg(A \vee \neg A)\} \vdash \neg\neg A$, by $(\neg I)$, from (v) and (vii).

(ix) $\vdash \neg\neg(A \vee \neg A)$, by $(\neg I)$, from (iv) and (viii).

(x) $\vdash (A \vee \neg A)$, by (DNE), from (ix).

The principle $(\theta \vee \neg\theta)$ is sometimes called the *law of excluded middle*. It is not valid in intuitionistic logic.

Let $\theta, \neg\theta$ be a pair of contradictory opposites, and let ψ be any formula at all. By (As) we have $\{\theta, \neg\theta, \neg\psi\} \vdash \theta$ and $\{\theta, \neg\theta, \neg\psi\} \vdash \neg\theta$. So by $(\neg I)$, $\{\theta, \neg\theta\} \vdash \neg\neg\psi$. So, by (DNE) we have $\{\theta, \neg\theta\} \vdash \psi$. That is, anything at all follows from a pair of contradictory opposites. Some logicians introduce a rule to codify a similar inference:

If $\Gamma_1 \vdash \theta$ and $\Gamma_2 \vdash \neg\theta$, then for any formula ψ, $\Gamma_1, \Gamma_2 \vdash \psi$

The inference is sometimes called *ex falso quodlibet*. Some call it "$\neg$-elimination", but perhaps this stretches the notion of "elimination" a bit. We do not officially include *ex falso quodlibet* as a separate rule in *D*, but as will be shown below (Theorem 10), each instance of it is derivable.

Some logicians object to *ex falso quodlibet*, on the ground that the formula ψ may be *irrelevant* to any of the premises in Γ. Suppose, for example, that one starts with some premises Γ about human nature and facts about certain people, and then deduces both the sentence "Clinton had extra-marital sexual relations" and "Clinton did not have extra-marital sexual relations". One can surely conclude that there is something wrong with premises Γ. But should we be allowed to then deduce *anything at all* from Γ? Should we be allowed to deduce "The economy is sound"?

Deductive systems that demur from *ex falso quodlibet* are part of *relevance logic*. See Anderson and Belnap [1975], Anderson, Belnap, and Dunn [1992], and Tennant [1987]. Deep philosophical issues concerning the nature of logical consequence are involved. Far be it for an article in a philosophy encyclopedia to avoid philosophical issues, but space considerations preclude a fuller treatment of this issue here. Suffice it to note that the inference is sanctioned in systems of *classical logic*, the subject of this article. It is essential to establishing the balance between the deductive system and the semantics (see §5 below).

The next pieces of *D* are the clauses for the quantifiers. Let θ be a formula, *v* a variable, and *t* a term (i.e., a variable or a constant). Then define θ(*v*|*t*) to be the result of substituting *t* for each *free*occurrence of *v* in θ. So, if θ is (*Qx* & ∃*xPxy*), then θ(*x*|*c*) is (*Qc*& ∃*xPxy*). The last occurrence of *x* is not free (but recall that we avoid using formulas like this).

We have one other nicety to attend to. Suppose that v_1 and v_2 are variables. It may happen that some of the substituted instances ofv_2 are bound in θ(v_1|v_2). When this happens, we say that there is a *clash* of the variables. Suppose, for example, that θ is ∃*y*(¬*x* =*y*), and so θ(*x*|*y*) is ∃*y*(¬*y* = *y*). We say that a term *t* is *free for* a variable *v* in θ if either *t* is a constant or there is no clash of variables in θ(*v*|*t*). The idea is that no substituted instance of *t*should become a bound variable in θ(*v*|*t*).

A formula in the form ∀*v* θ is an analogue of the English "for every *v*, θ holds". So one should be able to infer θ(*v*|*t*) from ∀*v* θ:

(∀E) If Γ ⊢ ∀*v* θ, then Γ ⊢ θ(*v*|*t*), provided that *t* is free for *v* in θ.

The idea here is that if ∀*v* θ is true, then θ should hold of *t*, no matter what *t* is. We can illustrate the restriction on (∀E) as follows: The sentence ∀*x* ∃ *y*(¬*x*=*y*) corresponds to an assertion that for every object *x*, there is an object different from *x*. This is a coherent, plausible assertion. It is true if and only if the universe has at least two objects. It should follow that no matter what object *t* may be, something is different from *t*. However, if we were allowed to substitute the variable *y* for *x*, we would conclude ∃*y*(¬*y*=*y*), which says that there is something which is different from itself, a blatant falsehood.

The introduction clause for the universal quantifier is a bit more complicated. Suppose that a formula θ has a variable *v* free, and that θ has been deduced from a set of premises Γ. If the variable *v*does not occur free in any member of Γ, then θ will hold no matter which object *v* may denote. That is, ∀*v* θ follows.

(∀I) If Γ ⊢ θ and the variable *v* does not occur free in any member of Γ, then Γ ⊢ ∀*v*θ.

This introduction rule corresponds to a common inference in mathematics. Suppose that a mathematician says "let *n* be a natural number" and goes on to show that *n* has a certain property *P*, without assuming anything

about n (except that it is a natural number). She then reminds the reader that n is "arbitrary", and concludes that P holds for *all* natural numbers. The condition that the variable v not occur in any premise is what guarantees that it is indeed "arbitrary". It could be any object, and so anything we conclude about it holds for all objects.

The existential quantifier is an analogue of the English expression "there exists", or perhaps just "there is". If we have established (or assumed) that a given object t has a given property, then it follows that there is something that has that property. Again, we have to be careful with the syntax, and avoid clashes of variables.

($\exists$**I**) If $\Gamma \vdash \theta$, then $\Gamma \vdash \exists v \, \theta'$, where θ' is obtained from θ by substituting the variable v for zero or more occurrences of a term t, provided that (1) if t is a variable, then all of the replaced occurrences of t are free in θ, and (2) all of the substituted occurrences of v are free in θ'.

The provision (1) keeps us from replacing bound variables. The provision (2) comes up only if v is bound by another quantifier in θ. As noted above, we avoid such formulas (since they appear to bind the same occurrence twice).

The elimination rule for $\exists$ is not quite as simple:

($\exists$**E**) If $\Gamma_1 \vdash \exists v \, \theta$ and $\Gamma_2, \theta \vdash \phi$, then $\Gamma_1, \Gamma_2 \vdash \phi$, provided that v does not occur free in ϕ, nor in any member of Γ_2.

This elimination rule also corresponds to a common inference. Suppose that a mathematician asserts that there is a natural number with a given property P. She then says "let n be such a natural number, so that Pn", and goes on to establish a sentence ϕ, which does not mention the number n. If the derivation of ϕ does not invoke anything about n (other than the assumption that it has the given property P), then n could have been any number that has the property P. That is, n is an *arbitrary* number with property P. It does not matter which number n is. Since ϕ does not mention n, it follows from the assertion that something has property P. The provisions added to ($\exists$E) are to guarantee that x is "arbitrary".

As noted in the previous section, some authors introduce different letters for bound variables and (what amounts to) free variables. This makes the syntax

slightly more complex, but simplifies the provisions on some of the rules of inference. Writers of logic books often face tradeoffs like this.

The final items are the rules for the identity sign "=". The introduction rule is about a simple as can be:

(=I) $\Gamma \vdash t=t$, where t is any term.

This "inference" corresponds to the truism that everything is identical to itself. The elimination rule corresponds to a principle that if a is identical to b, then anything true of a is also true of b, again paying attention to clashes of variables.

(=E) If $\Gamma_1 \vdash t_1=t_2$ and $\Gamma_2 \vdash \theta$, then $\Gamma_1, \Gamma_2 \vdash \theta'$, where θ' is obtained from θ by replacing zero or more occurrences of t_1 with t_2, provided that no bound variables are replaced, and if t_2 is a variable, then all of its substituted occurrences are free.

The rule (=E) indicates a certain restriction in the expressive resources of our language. Suppose, for example, that Harry is identical to Donald (since his mischievous parents gave him two names). It would not follow from this and "Dick knows that Harry is wicked" that "Dick knows that Donald is wicked", for the reason that Dick might not know that Harry is identical to Donald. Contexts like this, in which identicals cannot safely be substituted for each other, are called "opaque". We assume that our language L1K= has no opaque contexts.

One final clause completes the description of the deductive system D:

(*) That's all folks. $\Gamma \vdash \theta$ only if θ follows from members of Γ by the above rules.

Again, this clause allows proofs by induction on the rules used to establish an inference. If a property of arguments holds of all instances of (As) and (=I), and if the other rules preserve the property, then every argument that is deducible in D enjoys the property in question.

Before moving on to the model theory for L1K=, we pause to note a few features of the deductive system.

Lemma 7. Suppose that $\Gamma \vdash_D \phi$, and let v' be a variable that does not occur free in ϕ or in any member of Γ. Assume that v' is free for v in ϕ and in every

member of Γ. Let Γ' be $\{\theta(v|v') \mid \theta \in \Gamma\}$. That is, Γ' is the result of replacing every free occurrence of a variable v with v' in every member of Γ. Then $\Gamma' \vdash_D \phi(v|v')$.

Proof: The proof of this lemma is tedious, but we give its essentials. We proceed by induction on the number of rules that were used to arrive at $\Gamma \vdash \phi$. Suppose that $n>0$ is a natural number, and that the lemma holds for any argument that was derived using fewer than n rules, and suppose that $\Gamma \vdash \phi$ using exactly n rules. If $n=1$, then the rule applied is either (As) or (=I). In this case, $\Gamma' \vdash \phi(v|v')$ by the same rule. If the last rule applied is (&I), then ϕ has the form $(\theta \, \& \, \psi)$, and we have $\Gamma_1 \vdash \theta$ and $\Gamma_2 \vdash \psi$, with $\Gamma = \Gamma_1, \Gamma_2$. We apply the induction hypothesis to the deductions of θ and ψ, and then apply (&I) to the result. If the last rule applied was (&E), we have two sub-cases, but they are symmetric. We have $\Gamma \vdash (\phi \, \& \, \psi)$. There are two slight complications here: the new variable v' may occur free in ψ and it may not be free for v in ψ. In either case, first pick a new variable u that does not occur (free or bound) in $(\phi \, \& \, \psi)$ or in any member of Γ. Now apply the induction hypothesis, substituting u for v' in the deduction $\Gamma \vdash (\phi \, \& \, \psi)$. Since v' does not occur free in ϕ or in any member of Γ, those formulas are left unchanged. The maneuver removes any free occurrences of v' from the subformula ψ. Now apply the induction hypothesis to the result, substituting v' for v, and then apply (&E). The remaining cases are similar.

Theorem 8. The rule of Weakening. If $\Gamma_1 \vdash \phi$ and $\Gamma_1 \subseteq \Gamma_2$, then $\Gamma_2 \vdash \phi$.

Proof: Again, we proceed by induction on the number of rules that were used to arrive at $\Gamma_1 \vdash \phi$. Suppose that $n>0$ is a natural number, and that the theorem holds for any argument that was derived using fewer than n rules. Suppose that $\Gamma_1 \vdash \phi$ using exactly n rules. If $n=1$, then the rule is either (As) or (=I). In these cases, $\Gamma_2 \vdash \phi$ by the same rule. If the last rule applied was (&I), then ϕ has the form $(\theta\&\psi)$, and we have $\Gamma_3 \vdash \theta$ and $\Gamma_4 \vdash \psi$, with $\Gamma_1 = \Gamma_3, \Gamma_4$. We apply the induction hypothesis to the deductions of θ and ψ, to get $\Gamma_2 \vdash \theta$ and $\Gamma_2 \vdash \psi$. and then apply (&I) to the result to get $\Gamma_2 \vdash \phi$. Most of the other cases are exactly like this. Slight complications arise only in the rules ($\forall$I) and ($\exists$E), because there we have to pay attention to the conditions for the rules. Starting with ($\exists$E), we have $\Gamma_3 \vdash \exists v\theta$ and $\Gamma_4, \theta \vdash \phi$, with Γ_1 being Γ_3, Γ_4, and v not free in ϕ, nor in any member of Γ_4. We apply the induction hypothesis to get $\Gamma_2 \vdash \exists v\theta$, and then ($\exists$E) to end up with $\Gamma_2 \vdash \phi$. Suppose that the last rule applied to get $\Gamma_1 \vdash \phi$ is ($\forall$I). So ϕ is a formula in the form $\forall v\theta$, and we have $\Gamma_1 \vdash \theta$ and the variable v does not

occur free in any member of Γ_1. The problem is that v may occur free in a member of Γ_2, and so we cannot just invoke the induction hypothesis and apply (∀I) to the result. Let v' be a variable that does not occur (free or bound) in θ or in any member of Γ_2, and let Γ' be the result of substituting v' for every free occurrence of v in Γ_2. Since v does not occur free in any member of Γ_1, we still have $\Gamma_1 \subseteq \Gamma$. The induction hypothesis gives us $\Gamma' \vdash \theta$, and now we apply (∀I) to get $\Gamma' \vdash \phi$. We now apply Lemma 7, substituting v for the new variable v'. The result is $\Gamma_2 \vdash \phi$.

Theorem 8 allows us to add on premises at will. It follows that $\Gamma \vdash \phi$ if and only if there is a subset $\Gamma' \subseteq \Gamma$ such that $\Gamma' \vdash \phi$. By clause (*), all derivations are established in a finite number of steps. So we have

Theorem 9. $\Gamma \vdash \phi$ if and only if there is a finite $\Gamma' \subseteq \Gamma$ such that $\Gamma' \vdash \phi$.

Theorem 10. The rule of ex falso quodlibet is a "derived rule" of *D*. That is, if $\Gamma_1 \vdash \theta$ and $\Gamma_2 \vdash \neg\theta$, then $\Gamma_1, \Gamma_2 \vdash \psi$, for any formula ψ.

Proof: Suppose that $\Gamma_1 \vdash \theta$ and $\Gamma_2 \vdash \neg\theta$. Then by Theorem 8, $\Gamma_1, \neg\psi \vdash \theta$, and $\Gamma_2, \neg\psi \vdash \neg\theta$. So by (¬I), $\Gamma_1, \Gamma_2 \vdash \neg\neg\psi$. By (DNE), $\Gamma_1, \Gamma_2 \vdash \psi$.

Theorem 11. The rule of Cut. If $\Gamma_1 \vdash \psi$ and $\Gamma_2, \psi \vdash \theta$, then $\Gamma_1, \Gamma_2 \vdash \theta$.

Proof: Suppose $\Gamma_1 \vdash \psi$ and $\Gamma_2, \psi \vdash \theta$. We proceed by induction on the number of rules used to establish $\Gamma_2, \psi \vdash \theta$. Suppose that n is a natural number, and that the theorem holds for any argument that was derived using fewer than n rules. Suppose that $\Gamma_2, \psi \vdash \theta$ was derived using exactly n rules. If the last rule used was (=I), then $\Gamma_1, \Gamma_2 \vdash \theta$ is also an instance of (=I). If $\Gamma_2, \psi \vdash \theta$ is an instance of (As), then either θ is ψ, or θ is a member of Γ_2. In the former case, we have $\Gamma_1 \vdash \theta$ by supposition, and get $\Gamma_1, \Gamma_2 \vdash \theta$ by Weakening (Theorem 8). In the latter case, $\Gamma_1, \Gamma_2 \vdash \theta$ is itself an instance of (As). Suppose that $\Gamma_2, \psi \vdash \theta$ was obtained using (&E). Then we have $\Gamma_2, \psi \vdash (\theta \& \phi)$. The induction hypothesis gives us $\Gamma_1, \Gamma_2 \vdash (\theta \& \phi)$, and (&E) produces $\Gamma_1, \Gamma_2 \vdash \theta$. The remaining cases are similar.

Theorem 11 allows us to chain together inferences. This fits the practice of establishing theorems and lemmas and then using those theorems and lemmas later, at will. The cut principle is, I think, essential to reasoning. In some logical systems, the cut principle is a deep theorem. The system here was designed to make the proof of Theorem 11 straightforward.

If $\Gamma \vdash_D \theta$, then we say that the formula θ is a *deductive consequence* of the set of formulas Γ, and that the argument $\langle\Gamma,\theta\rangle$ is *deductively valid*. A formula θ is a *logical theorem*, or a *deductive logical truth*, if $\vdash_D \theta$. That is, θ is a logical theorem if it is a deductive consequence of the empty set. A set Γ of formulas is *consistent* if there is no formula θ such that $\Gamma \vdash_D \theta$ and $\Gamma \vdash_D \neg\theta$. That is, a set is consistent if it does not entail a pair of contradictory opposite formulas.

Theorem 12. A set Γ is consistent if and only if there is a formula θ such that it is not the case that $\Gamma \vdash \theta$.

Proof: Suppose that Γ is consistent and let θ be any formula. Then either it is not the case that $\Gamma \vdash \theta$ or it is not the case that $\Gamma \vdash \neg\theta$. For the converse, suppose that Γ is inconsistent and let ψ be any formula. We have that there is a formula such that both $\Gamma \vdash \theta$ and $\Gamma \vdash \neg\theta$. By ex falso quodlibet (Theorem 10), $\Gamma \vdash \psi$.

Define a set Γ of formulas of the language L1K= to be *maximally consistent* if Γ is consistent and for every formula θ of L1K=, if θ is not in Γ, then Γ, θ is inconsistent. In other words, Γ is maximally consistent if Γ is consistent, and adding any formula in the language not already in Γ renders it inconsistent. Notice that if Γ is maximally consistent then $\Gamma \vdash \theta$ if and only if θ is in Γ.

Theorem 13. The Lindenbaum Lemma. Let Γ be any consistent set of formulas of L1K=. Then there is a set Γ' of formulas of L1K= such that $\Gamma \subseteq \Gamma'$ and Γ' is maximally consistent.

Proof: Although this theorem holds in general, we assume here that the set K of non-logical terminology is either finite or denumerably infinite (i.e., the size of the natural numbers, usually called $\aleph_0$). It follows that there is an enumeration θ_0, θ_1, … of the formulas of L1K=, such that every formula of L1K= eventually occurs in the list. Define a sequence of sets of formulas, by recursion, as follows: Γ_0 is Γ; for each natural number n, if Γ_n, θ_n is consistent, then let $\Gamma_{n+1} = \Gamma_n$, θ_n. Otherwise, let $\Gamma_{n+1} = \Gamma_n$. Let Γ' be the union of all of the sets Γ_n. Intuitively, the idea is to go through the formulas of L1K=, throwing each one into Γ' if doing so produces a consistent set. Notice that each Γ_n is consistent. Suppose that Γ' is inconsistent. Then there is a formula θ such that $\Gamma' \vdash \theta$ and $\Gamma' \vdash \neg\theta$. By Theorem 9 and Weakening (Theorem 8), there is finite subset Γ'' of Γ' such that $\Gamma'' \vdash \theta$ and $\Gamma'' \vdash \neg\theta$. Because Γ'' is finite, there is a natural number n such that every member of Γ'' is in Γ_n. So, by Weakening again, $\Gamma_n \vdash \theta$ and $\Gamma_n \vdash \neg\theta$. So Γ_n is inconsistent,

which contradicts the construction. So Γ′ is consistent. Now suppose that a formula θ is not in Γ′. We have to show that Γ′, θ is inconsistent. The formula θ must occur in the aforementioned list of formulas; say that θ is $θ_m$. Since $θ_m$ is not in Γ′, then it is not in $Γ_{m+1}$. This happens only if $Γ_m$, $θ_m$ is inconsistent. So a pair of contradictory opposites can be deduced from $Γ_m,θ_m$. By Weakening, a pair of contradictory opposites can be deduced from Γ′, $θ_m$. So Γ′, $θ_m$ is inconsistent. Thus, Γ′ is maximally consistent.

Notice that this proof uses a principle corresponding to the law of excluded middle. In the construction of Γ′, we assumed that, at each stage, either $Γ_n$ is consistent or it is not. Intuitionists, who demur from excluded middle, do not accept the Lindenbaum lemma (see Shapiro [1988]).

4. Semantics

Let *K* be a set of non-logical terminology. An *interpretation* for the language L1*K*= is a structure *M* = <*d*,*I*>, where *d* is a non-empty set, called the *domain-of-discourse*, or simply the *domain*, of the interpretation, and *I* is an *interpretation function*. Informally, the domain is what we interpret the language L1*K*= to be about. It is what the variables range over. The interpretation function assigns appropriate extensions to the non-logical terms. In particular,

If *c* is a constant in *K*, then *I*(*c*) is a member of the domain *d*.

If P^0 is a zero-place predicate letter in *K*, then $I(P^0)$ is a truth value, either truth or falsehood.

If Q^1 is a one-place predicate letter in *K*, then *I*(*Q*) is a subset of *d*. Intuitively, *I*(*Q*) is the set of members of the domain that the predicate *Q* holds of. If *Q* represents "red", then *I*(*Q*) is the set of red members of the domain.

If R^2 is a two-place predicate letter in *K*, then *I*(*R*) is a set of ordered pairs of members of *d*. Intuitively, *I*(*R*) is the set of pairs of members of the domain that the relation *R* holds between. If *R* represents "love", then *I*(*R*) is the set of pairs <*a*,*b*> such that *a* and *b* are the members of the domain for which *a* loves *b*.

In general, if S^n is an *n*-place predicate letter in *K*, then *I*(*S*) is a set of ordered *n*-tuples of members of *d*.

Define s to be a *variable-assignment*, or simply an *assignment*, on an interpretation M, if s is a function from the variables to the domain d of M. The role of variable-assignments is to assign denotations to the *free* variables of open formulas. (In a sense, the quantifiers determine the "meaning" of the bound variables.) Logical systems that dispense with free variables do not need variable-assignments, but some other device is employed.

We now define a relation of *satisfaction* between interpretations, variable-assignments, and formulas of L1K=. If ϕ is a formula ofL1K=, M is an interpretation for L1K=, and s is a variable-assignment on M, then we write $M,s \vDash \phi$ for M *satisfies ϕ under the assignment s*. The idea is that $M,s \vDash \phi$ is an analogue of "ϕ comes out true when interpreted as in M via s".

Let t be a term of L1K=. We define the *denotation* of t in Munder s, in terms of the interpretation function and variable-assignment:

If c is a constant, then $D_{M,s}(c)$ is $I(c)$, and if v is a variable, then $D_{M,s}(v)$ is $s(v)$.

That is, the interpretation M assigns denotations to the constants, while the variable-assignment assigns denotations to the (free) variables. If the language contained function symbols, the denotation function would be defined by recursion.

We now proceed by recursion on the complexity of the formulas of L1K=.

If t_1 and t_2 are terms, then $M,s \vDash t_1 = t_2$ if and only if$D_{M,s}(t_1)$ is the same as $D_{M,s}(t_2)$.

This is about as straightforward as it gets. An identity $t_1 = t_2$comes out true if and only if the terms t_1 and t_2 denote the same thing.

If P^0 is a zero-place predicate letter in K, then $M,s \vDash P$ if and only if $I(P)$ is truth.

If S^n is an n-place predicate letter in K and t_1, … , t_n are terms, then $M,s \vDash St_1 \ldots t_n$ if and only if the n-tuple $<D_{M,s}(t_1), \ldots , D_{M,s}(t_n)>$ is in $I(S)$.

This takes care of the atomic formulas. We now proceed to the compound formulas of the language, following the meanings of the English counterparts of the logical terminology.

$M,s \vDash \neg\theta$ if and only if it is not the case that $M,s \vDash \theta$.

$M,s \vDash (\theta \& \psi)$ if and only if both $M,s \vDash \theta$ and $M,s \vDash \psi$.

$M,s \vDash (\theta \vee \psi)$ if and only if either $M,s \vDash \theta$ or $M,s \vDash \psi$.

$M,s \vDash (\theta \rightarrow \psi)$ if and only if either it is not the case that $M,s \vDash \theta$, or $M,s \vDash \psi$.

$M,s \vDash \forall v\theta$ if and only if $M,s' \vDash \theta$, for every assignment s' that agrees with s except possibly at the variable v.

The idea here is that $\forall v\theta$ comes out true if and only if θ comes out true no matter what is assigned to the variable v. The final clause is similar.

$M,s \vDash \exists v\theta$ if and only if $M,s' \vDash \theta$, for some assignment s' that agrees with s except possibly at the variable v.

So $\exists v\theta$ comes out true if there is an assignment to v that makes θ true.

Theorem 6, unique readability, assures us that this definition is coherent. At each stage in breaking down a formula, there is exactly one clause to be applied, and so we never get contradictory verdicts concerning satisfaction.

As indicated, the role of variable-assignments is to give denotations to the free variables. We now show that variable-assignments play no other role.

Theorem 14. For any formula θ, if s_1 and s_2 agree on the free variables in θ, then $M,s_1 \vDash \theta$ if and only if $M,s_2 \vDash \theta$.

Proof: We proceed by induction on the complexity of the formula θ. The theorem clearly holds if θ is atomic, since in those cases only the values of the variable-assignments at the variables in θ figure in the definition. Assume, then, that the theorem holds for all formulas less complex than θ. And suppose that s_1 and s_2 agree on the free variables of θ. Assume, first, that θ is a negation, $\neg\psi$. Then, by the induction hypothesis, $M,s_1 \vDash \psi$ if and only if $M,s_2 \vDash \psi$. So, by the clause for negation, $M,s_1 \vDash \neg\psi$ if and only if $M,s_2 \vDash \neg\psi$. The cases where the main connective in θ is a binary connectives are also straightforward. Suppose that θ is $\exists v\psi$, and that $M,s_1 \vDash \exists v\psi$. Then there is an assignment s_1' that agrees with s_1 except possibly at v such that $M,s_1' \vDash \psi$. Let s_2' be the assignment that agrees with s_2 on the free variables not in and agrees with s_1' on the others. Then, by the induction hypothesis, $M,s_2' \vDash \psi$. Notice that s_2' agrees with s_2 on

every variable except possibly *v*. So $M,s_2 \vDash \exists v\psi$. The converse is the same, and the case where θ begins with a universal quantifier is similar.

Recall that a sentence is a formula with no free variables. So by Theorem 14, if θ is a sentence, and s_1, s_2, are any two variable-assignments, then $M,s_1 \vDash θ$ if and only if $M,s_2 \vDash θ$. So we can just write $M \vDash θ$ if $M,s \vDash θ$ for some, or all, variable-assignments *s*.

Suppose that $K' \subseteq K$ are two sets of non-logical terms. If $M = <d,I>$ is an interpretation of L1*K=*, then we define the *restriction* of *M* to L1*K'* be the interpretation $M' = <d,I'>$ such that *I'* is the restriction of *I* to *K'*. That is, *M* and *M'* have the same domain and agree on the non-logical terminology in *K'*. A straightforward induction establishes the following:

Theorem 15. If *M'* is the restriction of *M* to L1*K'*, then for every formula θ of L1*K'*, if *s* is any variable-assignment, $M,s \vDash θ$ if and only if $M',s \vDash θ$.

Theorem 16. If two interpretations M_1, M_2 have the same domain and agree on the non-logical terminology of a formula θ, then if *s* is any variable-assignment, $M_1,s \vDash θ$ if and only if $M_2,s \vDash θ$.

In short, the satisfaction of a formula θ only depends on the domain of discourse, the interpretation of the non-logical terminology in θ, and the assignments to the free variables in θ.

We say that an argument <Γ,θ> is *semantically valid*, or just *valid*, written $Γ \vDash θ$, if for every interpretation *M* of the language and any variable-assignment *s* on *M*, if $M,s \vDash ψ$, for every member ψ of Γ, then $M,s \vDash θ$. If $Γ \vDash θ$, we also say that θ is a *logical consequence*, or *semantic consequence*, or *model-theoretic consequence* of Γ. The definition corresponds to the informal idea that an argument is valid if it is not possible for its premises to all be true and its conclusion false. Our definition of logical consequence also sanctions the common thesis that a valid argument is truth-preserving--to the extent that satisfaction represents truth. Officially, an argument in L1*K=* is valid if its conclusion comes out true under every interpretation of the language in which the premises are true. Validity is the model-theoretic counterpart to deducibility.

A formula θ is *logically true*, or *valid*, if $M,s \vDash θ$, for every interpretation *M* and assignment *s*. A formula is logically true if and only if it is a consequence of the

empty set. If θ is logically true, then for any set Γ of formulas, Γ ⊨ θ. Logical truth is the model-theoretic counterpart of theoremhood.

A formula θ is *satisfiable* if there is an interpretation M and a variable-assignment s on M such that $M,s ⊨ θ$. That is, θ is satisfiable if there is an interpretation and assignment that satisfies it. A set Γ of formulas is satisfiable if there is an interpretation M and a variable-assignment s on M such that $M,s ⊨ θ$, for every formula θ in Γ. If Γ is a set of sentences and if $M ⊨ θ$ for each sentence θ in Γ, then we say that M is a *model of* Γ. So a set of sentences is satisfiable if it has a model. Satisfiability is the model-theoretic counterpart to consistency.

Notice that Γ ⊨ θ if and only if the set Γ,¬θ is not satisfiable. It follows that if a set Γ is not satisfiable, then if θ is any formula, Γ ⊨ θ. This is a model-theoretic counterpart to ex falso quodlibet (see Theorem 10). We have the following, as an analogue to Theorem 12:

Theorem 17. Let Γ be a set of formulas. The following are equivalent: (a) Γ is satisfiable; (b) there is no formula θ such that both Γ ⊨ θ and Γ ⊨ ¬θ; (c) there is some formula ψ such that it is not the case that Γ ⊨ ψ.

Proof: (a)⇒(b): Suppose that Γ is satisfiable and let θ be any formula. There is an interpretation M and assignment s such that $M,s ⊨ ψ$ for every member ψ of Γ. By the clause for negations, we cannot have both $M,s ⊨ θ$ and $M,s ⊨ ¬θ$. So either <Γ,θ> is not valid or else <Γ,¬θ> is not valid. (b)⇒(c): This is immediate. (c)⇒(a): Suppose that it is not the case that Γ ⊨ ψ. Then there is an interpretation M and an assignment s such that $M,s ⊨ θ$, for every formula θ in Γ and it is not the case that $M,s ⊨ ψ$. A fortiori, M,s satisfies every member of Γ, and so Γ is satisfiable.

5. Meta-theory

We now present some results that relate the deductive notions to their model-theoretic counterparts. The first one is probably the most straightforward. We motivated both the various rules of the deductive system D and the various clauses in the definition of satisfaction in terms of the meaning of the English counterparts to the logical terminology. So one would expect that an argument is deducible, or deductively valid, only if it is semantically valid.

Theorem 18. Soundness. For any formula θ and set Γ of formulas, if $\Gamma \vdash_D \theta$, then $\Gamma \vDash \theta$.

Proof: We proceed by induction on the number of clauses used to establish $\Gamma \vdash \theta$. So let n be a natural number, and assume that the theorem holds for any argument established as deductively valid with fewer than n steps. And suppose that $\Gamma \vdash \theta$ was established using exactly n steps. If the last rule applied was (=I) then θ is a formula in the form $t=t$, and so θ is logically true. A fortiori, $\Gamma \vDash \theta$. If the last rule applied was (As), then θ is a member of Γ, and so of course any interpretation and assignment that satisfies every member of Γ also satisfies θ. Suppose the last rule applied is (&I). So θ has the form $(\phi \& \psi)$, and we have $\Gamma_1 \vdash \phi$ and $\Gamma_2 \vdash \psi$, with $\Gamma = \Gamma_1, \Gamma_2$. The induction hypothesis gives us $\Gamma_1 \vDash \phi$ and $\Gamma_2 \vDash \psi$. Suppose that M,s satisfies every member of Γ. Then M,s satisfies every member of Γ_1, and so M,s satisfies ϕ. Similarly, M,s satisfies every member of Γ_2, and so M,s satisfies ψ. Thus, by the clause for "&" in the definition of satisfaction, M,s satisfies θ. So $\Gamma \vDash \theta$. Suppose the last clause applied was ($\exists$E). So we have $\Gamma_1 \vDash \exists v\psi$ and $\Gamma_2, \psi \vDash \theta$, where $\Gamma = \Gamma_1, \Gamma_2$, and v does not occur free in ψ, nor in any member of Γ_2. By the induction hypothesis, we have $\Gamma_1 \vdash \exists v\psi$ and $\Gamma_2, \psi \vDash \theta$. Let M be an interpretation and s an assignment such that M,s satisfies every member of Γ. Then M,s satisfies every member of Γ_1, and so $M,s \vDash \exists v\psi$. So there is an assignment s' that agrees with s on every variable except possibly v such that $M,s' \vDash \psi$. We have that M,s satisfies every member of Γ_2. Since v does not occur free in any member of Γ_2, and s agrees with s' on everything else, we have that M,s' satisfies every member of Γ_2, by Theorem 14. So $M,s' \vDash \theta$. Since v does not occur free in θ, and s agrees with s on everything else, we have that $M,s \vDash \theta$, also by Theorem 14. So, in this case, $\Gamma \vDash \theta$. Notice the role of the restrictions on ($\exists$E) here. The other cases are about as straightforward.

Corollary 19. Let Γ be a set of formulas. If Γ is satisfiable, then Γ is consistent.

Proof: Suppose that Γ is satisfiable. So let M be an interpretation and s an assignment such that M,s satisfies every member of Γ. Assume that Γ is inconsistent. Then there is a formula θ such that $\Gamma \vdash \theta$ and $\Gamma \vdash \neg\theta$. By soundness (Theorem 18), $\Gamma \vDash \theta$ and $\Gamma \vDash \neg\theta$. So we have that $M,s \vDash \theta$ and $M,s \vDash \neg\theta$. But this is impossible, given the clause for negation in the definition of satisfaction.

Even though the deductive system *D* and the model-theoretic semantics were developed with the meanings of the logical terminology in mind, one should not automatically expect the converse to soundness (or Corollary 19) to hold. For all we know so far, we may not have included enough rules of inference to deduce every valid argument. The converses to soundness and Corollary 19 are among the most interesting results in contemporary mathematical logic. We begin with the latter.

Theorem 20. Completeness. Gödel [1930]. Let Γ be a set of formulas. If Γ is consistent, then Γ is satisfiable.

Proof: The proof of completeness is rather complex. We only sketch it here. Let Γ be a consistent set of formulas of L1*K*=. Again, we assume for simplicity that the set *K* of non-logical terminology is either finite or countably infinite (although the theorem holds even if *K* is uncountable). The task at hand is to find an interpretation *M* and a variable-assignment *s* on *M*, such that *M,s* satisfies every member of Γ. Consider the language obtained from L1*K*= by adding a denumerably infinite stock of new individual constants $c_0, c_1, \ldots$ We stipulate that the constants, $c_0, c_1, \ldots$, are all different from each other and none of them occur in *K*. We build an interpretation of the language from the language itself, using some of the constants as members of the domain of discourse. Let $\theta_0, \theta_1, \ldots$ be an enumeration of the formulas of the expanded language, so that each formula occurs in the list eventually. Let *x* be any variable, and define a sequence $\Gamma_0, \Gamma_1, \ldots$ of sets of formulas (of the expanded language) by recursion as follows: $\Gamma_0 = \Gamma$; and $\Gamma_{n+1} = \Gamma_n, (\exists x \theta_n \to \theta_n(x|c_i))$, where c_i is the first constant in the above list that does not occur in θ_n or in any member of Γ_n. The underlying idea here is that if $\exists x \theta_n$ is true, then c_i is to be one such *x*. Let Γ be the union of the sets Γ_n.

I sketch a proof that Γ' is consistent. Suppose that Γ' is inconsistent. By Theorem 9, there is a finite subset of Γ that is inconsistent, and so one of the sets Γ_m is inconsistent. By hypothesis, $\Gamma_0 = \Gamma$ is consistent. Let *n* be the smallest number such that Γ_n is consistent, but $\Gamma_{n+1} = \Gamma_n, (\exists x \theta_n \to \theta_n(x|c_i))$ is inconsistent. By (¬I), we have that

(1) $\Gamma_n \vdash \neg(\exists x \theta_n \to \theta_n(x|c_i))$.

By *ex falso quodlibet* (Theorem 10), Γ_n, $\neg\exists x\theta_n$, $\exists x\theta_n \vdash \theta_n(x|c_i)$. So by ($\rightarrow$I), Γ_n, $\neg\exists x\theta_n \vdash (\exists x\theta_n \rightarrow \theta_n(x|c_i))$. From this and (1), we have $\Gamma_n \vdash \neg\neg\exists x\theta_n$, by ($\neg$I), and by (DNE) we have

(2) $\Gamma_n \vdash \exists x\theta_n$.

By (As), Γ_n, $\theta_n(x|c_i)$, $\exists x\theta_n \vdash \theta_n(x|c_i)$. So by ($\rightarrow$I), Γ_n, $\theta_n(x|c_i) \vdash (\exists x\theta_n \rightarrow \theta_n(x|c_i))$. From this and (1), we have $\Gamma_n \vdash \neg\theta_n(x|c_i)$, by ($\neg$I). Let v be a variable that does not occur (free or bound) in θ_n or in any member of Γ_n. By uniform substitution of v for c_i, we can turn the derivation of $\Gamma_n \vdash \neg\theta_n(x|c_i)$ into $\Gamma_n \vdash \neg\theta_n(x|v)$. By ($\forall$I), we have

(3) $\Gamma_n \vdash \forall v\neg\theta_n(x|v)$.

By (As) we have $\{\forall v\neg\theta_n(x|v),\theta_n\} \vdash \theta_n$ and by ($\forall$E) we have $\{\forall v\neg\theta_n(x|v), \theta_n\} \vdash \neg\theta_n$. So $\{\forall v\neg\theta_n(x|v), \theta_n\}$ is inconsistent. Let ϕ be any sentence of the language (so that ϕ has no free variables). By ex falso quodlibet (Theorem 10), we have that $\{\forall v\neg\theta_n(x|v),\theta_n\} \vdash \phi$ and $\{\forall v\neg\theta_n(x|v), \theta_n\} \vdash \neg\phi$. So with (2), we have that Γ_n, $\forall v\neg\theta_n(x|v) \vdash \phi$ and Γ_n, $\forall v\neg\theta_n(x|v) \vdash \neg\phi$, by ($\exists$E). By Cut (Theorem 11), $\Gamma_n \vdash \phi$ and $\Gamma_n \vdash \neg\phi$. So Γ_n is inconsistent, contradicting the assumption. So Γ' is consistent.

Applying the Lindenbaum Lemma (Theorem 13), let Γ'' be a maximally consistent set of sentences (of the expanded language) that contains Γ'. So, of course, Γ'' contains Γ. We define an interpretation M, and a variable-assignment s on M, such that M,s satisfies every member of Γ''.

If we did not have a sign for identity in the language, we would let the domain of M be the collection of new constants $\{c_0, c_1, \ldots \}$. But as it is, there may be a sentence in the form $c_i=c_j$, with $i\neq j$, in Γ''. If so, we cannot have both c_i and c_j in the domain of the interpretation. So we define the domain d of M to be the set $\{c_i \mid$ there is no $j<i$ such that $c_i=c_j$ is in $\Gamma''\}$. In other words, a constant c_i is in the domain of M if Γ'' does not declare it to be identical to an earlier constant in the list. Notice that for each new constant c_i, there is exactly one $j\leq i$ such that c_j is in d and the sentence $c_i=c_j$ is in Γ''.

We now define the interpretation function I. Let a be any constant in the expanded language. By (=I) and ($\exists$I), $\Gamma'' \vdash \exists x\ x=a$, and so $\exists x\ x=a \in \Gamma''$. By the construction of Γ', there is a sentence in the form ($\exists x\ x=a \rightarrow c_i=a$) in Γ''. We have

that $c_i=a$ is in Γ''. As above, there is exactly one c_j in d such that $c_i=c_j$ is in Γ''. Let $I(a)=c_j$. Notice that if c_i is a constant in the domain d, then $I(c_i)=c_i$. That is each c_i in d denotes itself.

Let P be a one-place predicate letter in K. Let $I(P)$ be the set of constants $\{c_i \mid c_i$ is in d and the formula Pc is in $\Gamma''\}$. Let R be a binary predicate letter in K. Let $I(R)$ be the set of pairs of constants $\{<c_i,c_j> \mid c_i$ is in d, c_j is in d, and the formula Rc_ic_j is in $\Gamma''\}$. Three-place predicates, etc. are interpreted similarly. In effect, I interprets the non-logical terminology as they are in Γ''.

The variable-assignment is similar. If v is a variable, then $s(v)=c_i$, where c_i is the first constant in d such that $c_i=v$ is in Γ''.

The final item in this proof is a tedious lemma that for every formula θ in the expanded language, $M,s \vDash \theta$ if and only if θ is in Γ''. This proceeds by induction on the complexity of θ. The case where θ is atomic follows from the definitions of M (i.e., the domain d and the interpretation function I) and the variable-assignment s. The other cases follow from the various clauses in the definition of satisfaction.

Since $\Gamma \subseteq \Gamma''$, we have that M,s satisfies every member of Γ. By Theorem 15, the restriction of M to the original language L1K= and s also satisfies every member of Γ. Thus Γ is satisfiable.

A converse to Soundness (Theorem 18) is a straightforward corollary:

Theorem 21. For any formula θ and set Γ of formulas, if $\Gamma \vDash \theta$, then $\Gamma \vdash_D \theta$.

Proof: Suppose that $\Gamma \vDash \theta$. Then there is no interpretation M and assignment s such that M,s satisfies every member of Γ but does not satisfy θ. So the set $\Gamma,\neg\theta$ is not satisfiable. By Completeness (Theorem 20), $\Gamma,\neg\theta$ is inconsistent. So there is a formula ϕ such that $\Gamma,\neg\theta \vdash \phi$ and $\Gamma,\neg\theta \vdash \neg\phi$. By ($\neg$I), $\Gamma \vdash \neg\neg\theta$, and by (DNE) $\Gamma \vdash \theta$.

Our next item is a corollary of Theorem 9, Soundness (Theorem 18), and Completeness:

Corollary 22. Compactness. A set Γ of formulas is satisfiable if and only if every finite subset of Γ is satisfiable.

Proof: If M,s satisfies every member of Γ, then M,ssatisfies every member of each finite subset of Γ. For the converse, suppose that Γ is not satisfiable. Then we show that some finite subset of Γ is not satisfiable. By Completeness (Theorem 20), Γ is inconsistent. By Theorem 9 (and Weakening), there is a finite subset $\Gamma'\subseteq\Gamma$ such that Γ' is inconsistent. By Corollary 19, Γ' is not satisfiable.

Soundness and completeness together entail that an argument is deducible if and only if it is valid, and a set of formulas is consistent if and only if it is satisfiable. So we can go back and forth between model-theoretic and proof-theoretic notions, transferring properties of one to the other. Compactness holds in the model theory because all derivations use only a finite number of premises.

Recall that in the proof of Completeness (Theorem 20), we made the simplifying assumption that the set K of non-logical constants is either finite or denumerably infinite. The interpretation we produced was itself either finite or denumerably infinite. Thus, we have the following:

Corollary 23. Löwenheim-Skolem Theorem. Let Γ be a satisfiable set of sentences of the language L1K=. If Γ is either finite or denumerably infinite, then Γ has a model whose domain is either finite or denumerably infinite.

In general, let Γ be a satisfiable set of sentences of L1K=, and let κ be the larger of the size of Γ and denumerably infinite. Then Γ has a model whose domain is at most size κ.

There is a stronger version of Corollary 23. Let $M_1=\langle d_1,I_1\rangle$ and $M_2=\langle d_2,I_2\rangle$ be interpretations of the language L1K=. Define $M_1$to be a *submodel* of M_2 if $d_1 \subseteq d_2$, $I_1(c) = I_2(c)$ for each constant$c$, and I_1 is the restriction of I_2 to d_1. For example, if R is a binary relation letter in K, then for all a,b in d_1, the pair $\langle a,b\rangle$ is in$I_1(R)$ if and only if $\langle a,b\rangle$ is in $I_2(R)$. If we had included function letters among the non-logical terminology, we would also require that d_1 be closed under their interpretations in M_2. Notice that ifM_1 is a submodel of M_2, then any variable-assignment on M_1 is also a variable-assignment on M_2.

Say that two interpretations $M_1=\langle d_1,I_1\rangle$, $M_2=\langle d_2,I_2\rangle$ are*elementarily equivalent* if one of them is a submodel of the other, and for any formula of the language and any variable-assignment s on the submodel, $M_1,s \vDash \theta$ if and only if $M_2,s \vDash \theta$.

Notice that if two interpretations are elementarily equivalent, then they satisfy the same sentences.

Theorem 25. Downward Löwenheim-Skolem Theorem. Let $M = <d,I>$ be an interpretation of the language L1K=. Let d_1 be any subset of d, and let κ be the maximum of the size of K, the size of d_1, and denumerably infinite. Then there is a submodel $M' = <d',I'>$ of M such that (1) d' is not larger than κ, and (2) M and M' are elementarily equivalent. In particular, if the set K of non-logical terminology is either finite or denumerably infinite, then any interpretation has an elementarily equivalent submodel whose domain is either finite or denumerably infinite.

Proof: Like completeness, this proof is complex, and we rest content with a sketch. The downward Löwenheim-Skolem theorem invokes the axiom of choice, and indeed, is equivalent to the axiom of choice. So let C be a choice function on the powerset of d, so that for each non-empty subset $e⊆d$, $C(e)$ is a member of e. We stipulate that if e is the empty set, then $C(e)$ is $C(d)$.

Let s be a variable-assignment on M, let θ be a formula of L1K=, and let v be a variable. Define the *v-witness of* θ*over s*, written $w_v(θ,s)$, as follows: Let q be the set of all elements $c∈d$ such that there is a variable-assignment s' on M that agrees with s on every variable except possibly v, such that $M,s' ⊨ θ$, and $s'(v)=c$. Then $w_v(θ,s) = C(q)$. Notice that if $M,s ⊨ ∃vθ$, then q is the set of elements of the domain that can go for v in θ. Indeed, $M,s ⊨ ∃vθ$ if and only if q is non-empty. So if $M,s ⊨ ∃vθ$, then $w_v(θ,s)$ (i.e., $C(q)$) is a chosen element of the domain that can go for v in θ. In a sense, it is a "witness" that verifies $M,s ⊨ ∃vθ$.

If e is a non-empty subset of the domain d, then define a variable-assignment s to be an *e-assignment* if for all variables u, $s(u)$ is in e. That is, s is an e-assignment if s assigns an element of e to each variable. Define $sk(e)$, the *Skolem-hull* of e, to be the set:

$e ∪ \{w_v(θ,s) \mid θ$ is a formula in L1K=, v is a variable, and s is an e-assignment$\}$.

That is, the Skolem-Hull of e is the set e together with every v-witness of every formula over every e-assignment. Roughly, the idea is to start with e and then throw in enough elements to make each existentially quantified formula true. But we cannot rest content with the Skolem-hull, however. Once we throw the "witnesses" into the domain, we need to deal with $sk(e)$ assignments. In effect,

we need a set which is its own Skolem-hull, and also contains the given subset d_1.

We define a sequence of non-empty sets e_0, e_1, … as follows: if the given subset d_1 of d is empty and there are no constants in K, then let e_0 be $C(d)$, the choice function applied to the entire domain; otherwise let e_0 be the union of d_1 and the denotations under I of the constants in K. For each natural number n, e_{n+1} is $sk(e_n)$. Finally, let d' be the union of the sets e_n, and let I' be the restriction of I to d'. Our interpretation is $M' = <d',I'>$.

Clearly, d_1 is a subset of d', and so M' is a submodel of M. Let κ be the maximum of the size of K, the size of d_1, and denumerably infinite. A calculation reveals that the size of d' is at most κ, based on the fact that there are at most κ-many formulas, and thus, at most κ-many witnesses at each stage. Notice, incidentally, that this calculation relies on the fact that a denumerable union of sets of size at most κ is itself at most κ. This also relies on the axiom of choice.

The final item is to show that M' is elementarily equivalent to M: For every formula θ and every variable-assignment s on M',

$M,s \vDash \theta$ if and only if $M',s \vDash \theta$.

We proceed by induction on the complexity of θ. If θ is atomic, then the definition of satisfaction entails the equivalence. So let θ be non-atomic, and assume that $M,s \vDash \psi$ if and only if $M',s \vDash \psi$, for all assignments s on M' and all formulas ψ less complex than θ. Let s be any such assignment. If the main connective of θ is the negation sign or a binary connective, then the induction hypothesis entails that $M,s \vDash \theta$ if and only if $M',s \vDash \theta$. The remaining cases are those in which θ begins with a quantifier, i.e., θ is either $\exists v\psi$ or $\forall v\psi$. Suppose that $M',s \vDash \exists v\psi$. Then there is a variable-assignment s' that agrees with s except possibly at v such that $M',s' \vDash \psi$. By the induction hypothesis, $M,s' \vDash \psi$ and so $M,s \vDash \exists v\psi$. The converse is a bit tricky, and amounts to showing that the Skolem-hull of d' is d'. Assume that $M,s \vDash \exists v\psi$. We are given that s is a variable-assignment on d'. Since there are only finitely many free-variables in ψ, let n be any natural number such that for all variables u that occur free in ψ, $s(u)$ is in e_n. Let s_1 be an e_n-assignment that agrees with s on all of the free variables in ψ. Then, by Theorem 14, $M,s_1 \vDash \exists v\psi$. Let c be $w_v(\theta,s_1)$, the *v-witness of θ over s_1*. Notice that c is in e_{n+1} and so c is in d'. Let s_1' agree with s_1, except possibly at v,

and let $s_1'(v)=c$. So s_1' is a variable-assignment on M'. By the definition of the witness function, $M,s_1' \vDash \psi$. By the induction hypothesis, $M',s_1' \vDash \psi$, and so $M',s_1 \vDash \exists v\psi$. By Theorem 14, $M',s \vDash \exists v\psi$. The final case, where θ has the form $\forall v\psi$, is similar.

Another corollary to Compactness (Corollary 22) is the opposite of the Löwenheim-Skolem theorem:

Theorem 26. Upward Löwenheim-Skolem Theorem. Let Γ be any set of formulas of L$1K=$, such that for each natural number n, there is an interpretation $M_n = <d_n,I_n>$, and an assignment s_n on M_n, such that d_n has at least n elements, and M_n,s_n satisfies every member of Γ. In other words, Γ is satisfiable and there is no finite upper bound to the size of the interpretations that satisfy every member of Γ. Then for any infinite cardinal κ, there is an interpretation $M=<d,I>$ and assignment s on M, such that the size of d is *at least* κ and M,s satisfies every member of Γ. In particular, if Γ is a set of sentences, then it has arbitrarily large models.

Proof: Add a collection of new constants $\{c_\alpha \mid \alpha<\kappa\}$, of size κ, to the language, so that if c is a constant in K, then c_α is different from c, and if $\alpha<\beta<\kappa$, then c_α is a different constant than c_β. Consider the set of formulas Γ' consisting of Γ together with the set $\{\neg c_\alpha=c_\beta \mid \alpha\neq\beta\}$. That is, Γ' consists of Γ together with statements to the effect that any two different new constants denote different objects. Let Γ'' be any finite subset of Γ', and let m be the number of new constants that occur in Γ''. Then expand the interpretation M_m to an interpretation M_m' of the new language, by interpreting each of the new constants in Γ'' as a different member of the domain d_m. By hypothesis, there are enough members of d_m to do this. One can interpret the other new constants at will. So M_m is a restriction of M_m'. By hypothesis (and Theorem 15), M'_m,s_m satisfies every member of Γ. Also M'_m,s_m satisfies the members of $\{\neg c_\alpha=c_\beta \mid \alpha\neq\beta\}$ that are in Γ''. So M'_m,s_m satisfies every member of Γ''. By compactness, there is an interpretation $M = <d,I>$ and an assignment s on M such that M,s satisfies every member of Γ'. Since Γ' contains every member of $\{\neg c_\alpha=c_\beta \mid \alpha\neq\beta\}$, the domain d of M must be of size at least κ, since each of the new constants must have a different denotation. By Theorem 15, the restriction of M to the original language L1$K=$ satisfies every member of Γ, with the variable-assignment s.

The proofs of the downward and upward Löwenheim-Skolem theorems can be combined to show that for any satisfiable set Γ of sentences, if there is no finite bound on the models of Γ, then for any infinite cardinal κ, there is a model of Γ whose domain has size *exactly* κ. Moreover, if M is any interpretation whose domain is infinite, then for any infinite cardinal κ, there is an interpretation M' whose domain has size exactly κ such that M and M' are elementarily equivalent.

These results indicate a weakness in the expressive resources of first-order languages like L1K=. No satisfiable set of sentences can guarantee that its models are all denumerably infinite, nor can any satisfiable set of sentences guarantee that its models are uncountable. So in a sense, first-order languages cannot express the notion of "denumerably infinite", at least not in the model theory.

Let A be any set of sentences in a first-order language L1K=, where K includes terminology for arithmetic, and assume that every member of A is true of the natural numbers. We can even let A be the set of all sentences in L1K= that are true of the natural numbers. Then A has uncountable models, indeed models of any infinite cardinality. Such interpretations are sometimes called *unintended*, or *non-standard* models of arithmetic. Let B be any set of first-order sentences that are true of the real numbers, and let C be any first-order axiomatization of set theory. Then if B and C are satisfiable (in infinite interpretations), then each of them has denumerably infinite models. That is, any first-order, satisfiable set theory or theory of the real numbers, has (unintended) models the size of the natural numbers. This is despite the fact that a sentence (seemingly) stating that the universe is uncountable is provable in most set-theories. This situation, known as the *Skolem paradox*, has generated much discussion, but we must refer the reader elsewhere for a sample of it.

Bibliography

Cited Works

- Anderson, A. and N. Belnap [1975], *Entailment: The logic of relevance and necessity I*, Princeton: Princeton University Press.
- Anderson, A. and N. Belnap, and M. Dunn [1992], *Entailment: The logic of relevance and necessity II*, Princeton: Princeton University Press.

- Corcoran, J. [1973], "Gaps between logical theory and mathematical practice", *The methodological unity of science*, ed. by M. Bunge, Dordrecht: Holland, D. Reidel, 23-50.
- Church, A. [1956], *Introduction to mathematical logic*, Princeton: Princeton University Press. Classic textbook.
- Davidson, D. [1984], *Inquiries into truth and interpretation*, Oxford: Clarendon Press.
- Dummett, M. [1977], *Elements of intuitionism*, Oxford: Oxford University Press.
- Gödel, K. [1930], "Die Vollständigkeit der Axiome des logischen Funktionenkalkuls", *Montatshefte für Mathematik und Physik 37*, 349-360; translated as "The completeness of the axioms of the functional calculus of logic", in van Heijenoort [1967], 582-591.
- Lycan, W. [1984], *Logical form in natural language*, Cambridge, Massachusetts: The MIT Press.
- Montague, R. [1974], *Formal philosophy*, ed. by R. Thomason, New Haven: Yale University Press.
- Quine, W. V. O. [1960], *Word and object*, Cambridge, Massachusetts: The MIT Press.
- Quine, W. V. O. [1986], *Philosophy of logic*, second edition, Englewood Cliffs, New Jersey: Prentice-Hall.
- Shapiro, S. [1998], "Logical consequence: models and modality", in *The philosophy of mathematics today*, edited by M. Schirn, Oxford: Oxford University Press, 131-156.
- Tennant, N. [1987], "Conventional necessity and the contingency of deduction", *Dialectica 41*, 79-95.

Further Reading

- Boolos, G., and R. Jeffrey [1989], *Computability and logic*, third edition, Cambridge, England: Cambridge University Press. Textbook on mathematical logic.
- Enderton, H. [1972], *A mathematical introduction to logic*, New York: Academic Press. Textbook in mathematical logic.
- Forbes, G. [1994], *Modern Logic*, Oxford: Oxford University Press. Elementary textbook.
- Mendelson, E. [1987], *Introduction to mathematical logic*, third edition, Princeton: van Nostrand. Textbook in mathematical logic.

- Shapiro, S. [1996], *The limits of logic: Second-order logic and the Skolem paradox*, *The international research library of philosophy*, Dartmouth Publishing Company, 1996. An anthology containing many of the significant later papers on the Skolem paradox.
- Van Heijenoort, J [1967], *From Frege to Gödel*, Cambridge, Massachusetts: Harvard University Press. An anthology containing many of the major historical papers on mathematical logic in the early decades of the twentieth century.