PHILOSOPHICAL LOGIC

John P. Burgess

PRINCETON UNIVERSITY PRESS
PRINCETON AND OXFORD

Published by Princeton University Press,
41 William Street, Princeton, New Jersey 08540
In the United Kingdom: Princeton University Press,
6 Oxford Street, Woodstock, Oxfordshire OX20 1TW

Library of Congress Cataloging-in-Publication Data
Burgess, John P., 1948–
Philosophical logic / John P. Burgess.
p.   cm.   — (Princeton foundations of contemporary philosophy)
Includes bibliographical references and index.
ISBN 978-0-691-13789-6 (hardcover : alk. paper)
1. Logic. I. Title.
BC71.B89 2009 160—dc22 2008050131

British Library Cataloging-in-Publication Data is available

This book has been composed in Archer and Minion Pro

Printed on acid-free paper. ∞
press.princeton.edu

Printed in the United States of America
10 9 8 7 6 5 4 3 2 1

Logic, whether classical or extra- or anti-classical, is concerned with form. (On this traditional view of the subject, the phrase "formal logic" is pleonasm and "informal logic" oxymoron.) An argument is *logically valid*, its conclusion is a *logical consequence* of its premises, its premises *logically imply* its conclusions—three ways of saying the same thing—if and only if the argument is an instance of a logically valid *form* of argument. In modern logic forms are represented using formulas. What the reader of an introductory textbook is introduced to—what it is assumed the reader of this book has been introduced to—is on the one hand the art of *formalizing* arguments, representing their forms using formulas, and on the other hand the science of evaluating arguments once formalized.

What logical forms *are*, and how they are related to linguistic forms, are deep and difficult questions not of philosophical logic but of philosophy *of* logic. They are questions about what logicians are doing when they are at work, not questions that have to be resolved before logicians get to work. Indeed, logicians never would get to work if they waited for consensus to be achieved on such questions.

Similarly for the question of what premises and conclusions *are*. Here they will be spoken of as sentences rather than "propositions." It will be left to be tacitly understood that in general it is only when taken in context that sentences are true or false, and that for sentences to count as "the same" for purposes of logical analysis in a given context they need not consist of exactly the same words in exactly the same order. With these understandings, the only difference between sentences and "propositions" of real importance for our purposes will be that sentences can change in truth value over time, whereas it is said that "propositions" cannot (so that when a sentence changes truth value over time it is by expressing different "propositions" at different times).

All branches of philosophical logic borrow heavily from classical logic. While some previous acquaintance with classical logic is assumed here, introductory textbooks differ greatly in their notation and terminology, and a rapid review of the basics is called for, if for no other reason than to fix the particular symbolism and vocabulary that will be used in *this* book. The remainder of this chapter is a bare summary statement of the most important definitions and results pertaining to classical sentential and predicate logic. The reader may skim it on first reading and refer back to it as needed.

1.4 Classical Sentential Logic: Formulas

At the level of sentential logic, formulas are built up from *sentence letters* $p_0, p_1, p_2, \ldots$, standing in the place of sentences not further analyzed, using *connectives* written $\neg$, $\wedge$, $\vee$, $\rightarrow$, and $\leftrightarrow$, pronounced "not," "and," "or," "if," and "if and only if" (henceforth abbreviated "iff"), but representing negation, conjunction, disjunction, the conditional, and the biconditional, however expressed. The sentence letters are the *atomic* formulas. If A is a formula, then $\neg A$ is a formula. If A and B are formulas, then $(A \wedge B)$ is a formula, and similarly for the other three connectives. (The parentheses are to prevent ambiguities of grouping; in principle they should always be written, in practice they are not written when no ambiguity will result from omitting them.)

And those are *all* the formulas. And because those are all, in order to show that all formulas have some property, it is enough to show that atomic formulas have it, that if a formula has it, so does its negation, and that if two formulas have it, so does their conjunction, and similarly for the other connectives. This method of proof is called *induction on complexity*. One can also define a notion for all formulas by the similar method of *recursion on complexity*.

To give an example of formalization, consider the following argument, in words (1)–(2) and symbols (3)–(4):

(1) Portia didn't go without Queenie also going; and Portia went.
(2) Therefore, Queenie went.
(3) $\neg(p \wedge \neg q) \wedge p$
(4) q

When formalizing arguments, turning words into formulas, it is convenient to have as many connectives as possible available; but when proving results *about* formulas it is convenient to have as few as possible, since then in proofs and definitions by induction or recursion one has fewer cases to consider. One gets the best of both worlds if one considers only $\neg$ and $\wedge$, say, as *primitive* or part of the official notation, and the others as *defined*, or mere unofficial abbreviations: $A \vee B$ for $\neg(\neg A \wedge \neg B)$, $A \rightarrow B$ for $\neg A \vee B$, $A \leftrightarrow B$ for $(A \rightarrow B) \wedge (B \rightarrow A)$.

1.5 Classical Sentential Logic: Models

A form of argument is logically valid iff in any instance in which all the premises are true, the conclusion is true. Here instances of a form are what one obtains by putting specific sentences in for the sentence letters, to obtain specific premises and a specific conclusion that may be true or false. But it really does not matter what the sentences substituted *are*, or what they *mean*, but only whether they are *true*. For the connectives are *truth-functional*, meaning that the *truth value*, true or false, of a compound formed using one of them depends only on the truth values of the components from which it is formed. Thus the truth values of the instances of premise (3) and conclusion (4) depend only on the truth values of the sentences substituted for p and q, and not their meaning or identity. A *model* for (part or all of) classical sentential logic is just an assignment of truth values, conveniently represented by one for truth and zero for falsehood, to (some or all of) the sentence letters. Thus a model represents all that really matters for purposes of logical evaluation about an instance, so that in evaluating arguments it is not necessary to consider instances, but only models.

The extension of a model's assignment of truth values to all formulas (or if only some sentence letters have been assigned values, to all formulas in which only those sentence letters occur) is defined by recursion on complexity. The value of a negation is the opposite of the value of what is negated, and the value of a conjunction is the minimum of the values of what are conjoined. Writing $V \vDash A$ to indicate that model V makes formula A true, we have the following (wherein (7) and (8) follow from (5) and (6) and the definitions of $\vee$ and $\rightarrow$ in terms of $\neg$ and $\wedge$):

(5)	$V \vDash \neg A$	iff	not $V \vDash A$
(6)	$V \vDash A \wedge B$	iff	$V \vDash A$ and $V \vDash B$
(7)	$V \vDash A \vee B$	iff	$V \vDash A$ or $V \vDash B$
(8)	$V \vDash A \rightarrow B$	iff	$V \vDash B$ if $V \vDash A$

The argument from premises $A_1, A_2, \ldots, A_n$ to conclusion B is valid, the conclusion is a consequence or implication of the premises, iff every model (for any part of the formal language large enough to include all the sentence letters occurring in the relevant formulas) that makes the premises true makes the conclusion true. There is a separate terminology for two "degenerate" cases. If no model makes all of $A_1, A_2, \ldots, A_n$ true, then they are called jointly *unsatisfiable*, and otherwise jointly *satisfiable* (with "jointly" superfluous when $n = 1$). Like the notion of consequence, the notion of satisfiability makes sense for infinite as well as finite sets. If every model makes B true, it is called *valid*, and otherwise *invalid*. Note that if one formula is the negation of another, one of the two will be valid iff the other is unsatisfiable, and satisfiable iff the other is invalid.

Actually, the general notions of consequence and unsatisfiability, at least for finite sets, can be reduced to the special case of validity of a *formula*, by considering the formulas

(9) $\neg(A_1 \wedge A_2 \wedge \ldots \wedge A_n \wedge \neg B)$
(10) $\neg(\mathrm{A}_1 \wedge \mathrm{A}_2 \wedge \ldots \wedge \mathrm{A}_n)$

For the argument from the A_i to B is valid or invalid according as the formula (9), called its *leading principle*, is valid or invalid, and the A_i are satisfiable or unsatisfiable according as the formula (10) is invalid or valid. Two formulas A and B are *equivalent* iff each is a consequence of the other, or what comes to the same thing, iff the biconditional $A \leftrightarrow B$ is valid.

The valid formulas of classical sentential logic are called *tautologically* valid or simply *tautologies*; with other logics, *tautologies* mean not valid formulas of that logic but formulas of that logic that are substitution instances of valid formulas of classical sentential logic; *countertautologies* are formulas whose negations are tautologies. The term *tautological consequence* or *tautological implication* is used similarly.

1.6 Classical Sentential Logic: Decidability

When the intuitive but vague notion of "instance" is replaced by the technical but precise notion of "model," the need to check *infinitely* many cases is reduced to the need to check *finitely* many. In (3) and (4), for example, though there are infinitely many instances, or pairs of sentences that might be substituted for the sentence letters, there are only four models, or pairs of truth values that such sentences might have.

The result is that classical sentential logic is *decidable*. There is a *decision procedure* for validity, a mechanical procedure—a procedure such as in principle could be carried out by a computing machine—that will in all cases in a finite amount of time tell us whether a given formula is valid or invalid, satisfiable or unsatisfiable, namely, the procedure of checking systematically through all possible models. (The method of truth tables expounded in most introductory textbooks is one way of displaying such a systematic check.) It is easily checked that the argument (3)–(4) is valid (though it represents a form of argument rejected both by relevantists and by intuitionists).

1.7 Classical Predicate Logic: Formulas

There are many arguments that cannot be represented in classical sentential logic, above all arguments that turn on *quantification*, on statements about *all* or *some*. Classical predicate logic provides the means to formalize such arguments.

The notion of *formula* for predicate logic is more complex than it was for sentential logic. The basic symbols include, to begin with, *predicate letters* of various kinds: one-place predicate letters ${}^1P_0$, ${}^1P_1$, ${}^1P_2$, … , two-place predicate letters ${}^2P_0$, ${}^2P_1$, ${}^2P_2$, … , and so on. There are also the *variables* x_0, x_1, x_2, … , and an *atomic* formula now is a k-place predicate followed by k variables. Sometimes a special two-place predicate symbol $=$ for *identity* is included. It is written *between* its two variables (and its negation is abbreviated $\neq$). Formulas can be negated and conjoined as in sentential logic, but now a formula A can also be universally or existentially quantified with respect to any variable x_i, giving $\forall x_i A$ and $\exists x_i A$. However, just as disjunction was not really needed given negation and conjunction, so existential quantification is not really needed given negation and universal quantification, since $\exists x_i A$ can be taken to be an abbreviation for $\neg\forall x_i \neg A$.

To give an example, here is an argument and its formalization in classical predicate logic:

(11) All quarterlies are periodicals.
(12) Therefore, anyone who reads a quarterly reads a periodical.
(13) $\forall x(Qx \to Px)$
(14) $\forall y(\exists x(Qx \wedge Ryx) \to \exists x(Px \wedge Ryx))$

An important distinction, defined by recursion on complexity, is that between free and bound occurrences of a variable in a formula. All occurrences of variables in an atomic formula are free. The free occurrences of variables in the negation of a formula are those in the formula itself, and the free occurrences of variables in a conjunction of two formulas are those in the two formulas themselves. In a quantification $\forall x_i A$ on x_i the free occurrences of variables other than x_i are those in A, while all occurrences of x_i are bound rather than free. Formulas where every occurrence of every variable is bound are called *closed*; others, *open*. In a quantification $\forall yA$ on y, the free variables in A are said to be within the *scope* of the initial quantifier. We say y is *free for x in A* iff no free occurrence of x is within the scope of a quantification on y. In that case we write $A(y/x)$ for the result of replacing each free occurrence of x in A by y.

1.8 Classical Predicate Logic: Models

The notion of *model* for predicate logic, like the notion of formula, is more complex than it was for sentential logic. The idea is that no more matters for the truth values of premises and conclusion in any instance are what things are being spoken of, and which of the predicates substituted for the predicate letters are true of which of those things. What the predicates substituted for the predicate letters *are*, or what they *mean*, does not matter.

To specify a model $\boldsymbol{U}$ for (some or all of) the formal language of classical predicate logic we must specify its *universe* U, and also for (some or all of) the k-place predicate letters which things in U they are true of. This latter information can be represented in the form of a *denotation* function assigning as denotation to each one-place predicate letter ${}^1P_i$ a set ${}^1P_i^U$ of elements of U, assigning to each two-place predicate letter ${}^2P_i$ as denotation a set ${}^2P_i^U$ of pairs of elements of U, and so on. If identity is present, then $=^U$ is required to be the genuine identity relation on the universe U, which as a set of pairs is just $\{(u, u): u \in U\}$. Thus a model consists of a set of things and some distinguished relations among them (sets being counted as one-place relations, sets of pairs as two-place relations, and so on).

The notion of *truth* of a formula in a model for the formal language of classical predicate logic is also more complex. Its definition is one of the centerpieces of a good introductory course in logic, "Tarski's theory of truth." The notion of truth is applicable only to closed formulas, but to define it we must define a more general notion of *satisfaction* applicable to open formulas. Intuitively, a formula $A(y_1, \dots, y_k)$ with no more than the k free variables displayed is satisfied by a k-tuple $(u_1, \dots, u_k)$ of elements of U, or in symbols,

(15) $\boldsymbol{U} \vDash A(y_1, \dots, y_k)\ [u_1, \dots, u_k]$

iff the formula A is true when each free variable y_i is taken to stand for the corresponding element u_i. Truth is then simply the special case $k = 0$ of satisfaction. For present purposes we can work with this intuitive understanding, and there will be no need to recall the full technical definition, but it may be said that where $A(x)$ has just the one free variable, the analogues of (6) and (7) read as follows:

(16) $\boldsymbol{U} \vDash \forall x A(x)$ iff $\boldsymbol{U} \vDash A(x)[u]$ for all u in U
(17) $\boldsymbol{U} \vDash \exists x A(x)$ iff $\boldsymbol{U} \vDash A(x)[u]$ for some u in U

An argument is valid, its conclusion is a consequence or implication of its premises, iff every model (of a large enough part of the formal language to include all the predicate letters occurring in the relevant formulas) that makes the premise true makes the conclusion true. The notions of (un)satisfiability for a set of formulas, and (in)validity for a single formula, and equivalence of two formulas can then be introduced just as in sentential logic. While all these notions, involving as they do the notion of truth, in the first instance make sense only for closed formulas, they can be extended to open formulas. Thus $A(x, y, z)$ is valid iff it is satisfied by every (u, v, w) in every model, and satisfiable iff it is satisfied by some (u, v, w) in some model, or equivalently, is valid iff its *universal closure* $\forall x \forall y \forall z A(x, y, z)$ is valid, and satisfiable iff its *existential closure* $\exists x \exists y \exists z A(x, y, z)$ is satisfiable.

It has been indicated above that no more matters for the truth of a closed formula in a model than what objects are in the domain of the model and what distinguished relations among them the predicates of the language denote. But in fact a great deal less matters. All that really matters is the *number* of elements in the domain of the model, and the *pattern* of distinguished relations among them. For if we replace a model by another with the same number of elements and the same pattern of distinguished relations—the technical expression is: with an *isomorphic* model—then exactly the same closed formulas will be true. For instance, suppose we have a language with two one-place predicates *P* and *Q* and one two-place predicate *R*, and a model *M* with universe {♠, ♥, ♦, ♣} and with the denotations of *P* and *Q* being {♠, ♥} and {♦, ♣}, and the denotation of *R* being {(♠, ♥), (♥, ♦), (♦, ♣)}. If we replace it by the model *M′* with universe {1, 2, 3, 4} and with the denotations of *P* and *Q* being {1, 2} and {3, 4} and the denotation of *R* being {(1, 2), (2, 3), (3, 4)}, then exactly the same closed formulas will be true. In practice there is no need to consider any but *mathematical* models, models whose universes consist of mathematical objects, since every model is isomorphic to one of these. (In principle there would be no need to consider any but models whose universes are subsets of the set of natural numbers, though this fact depends on the Löwenheim-Skolem theorem, a result usually not covered in introductory texts.)

1.9 Classical Predicate Logic: Undecidability

According to *Church's theorem*, whose establishment is a major goal in intermediate-level textbooks, whereas for sentential logic there is a *decision procedure*, or method for *determining whether* a given formula is valid, for predicate logic there is none. Hence one looks for the next best thing, a *proof procedure*, or method for *demonstrating that* a given formula is valid, when it is.

Every introductory text presents some proof procedure or other, but hardly any two the same one, and there are several formats for proof procedures ("axiomatic," "natural deduction," "sequent calculus," "tableaux," "trees"), all quite different in appearance. Yet all styles have some features in common. With all, a *formal proof* or *demonstration* is some kind of finite array of symbols, and it is decidable whether or not a given finite array of symbols is a proof of a given formula.

The format of an *axiomatic*-style proof procedure, for instance, is as follows. Certain kinds of formulas are admitted as *axioms* and certain kinds of inferences from premise formulas to a conclusion formula are admitted as (*primitive*) *rules*. A proof (of a given formula) is a sequence of formulas (the last being the given formula) in which every formula or *step* either is an axiom or follows from earlier steps by a rule. In practice when proofs of this kind are presented, as they will be in later chapters, the steps are not just listed but annotated (using obvious abbreviations), with each step numbered on the left and marked on the right either as an axiom or as following from specified earlier steps by a specified rule.

For any style for proof procedure, a formula is *demonstrable* or a *theorem* iff there exists some formal proof or demonstration of it; otherwise it is *indemonstrable*. There are two related notions. Conclusion B is *deducible* from premises A_i iff the formula (9) is demonstrable, and the A_i are jointly *inconsistent* iff the formula (10) is demonstrable. We define B to be *deducible* from an infinite set iff it is deducible from some finite subset, and define an infinite set to be *inconsistent* iff some finite subset is.

And for any style of proof procedure, there are two results to be established for it, namely, that every demonstrable formula is valid and every valid formula is demonstrable. The task of establishing the first result, *soundness*, is generally tedious but routine. The second result, *completeness*, is another of the centerpieces of a good introductory course in logic, "Gödel's completeness theorem." Since the relationship of deducibility and consistency to demonstrability parallels the relationship of consequence and satisfiability to validity, the coincidence of validity with demonstrability yields the coincidence of the other *alethic* or truth-related notions, consequence and unsatisfiability, with their *apodictic* or proof-related counterparts, deducibility and inconsistency. (The coincidence of consequence and unsatisfiability with deducibility and inconsistency in the case of infinite sets of formulas depends on the compactness theorem, a result often not covered in introductory texts.)