

Propositional Logic Review

John MacFarlane

January 24, 2011

The task of describing a logical system comes in three parts:

Grammar Describing what counts as a formula.

Semantics Defining truth in a model (and, derivatively, logical consequence and related notions).

Proofs Describing what counts as a proof.

1 Grammar

- A *propositional constant* (a capital letter, possibly with a numerical subscript) is a formula.
- \perp is a formula.
- If p and q are formulas, then $(p \vee q)$, $(p \wedge q)$, $(p \supset q)$, $(p \equiv q)$, and $\neg p$ are formulas.
- Nothing else is a formula.

You might have used different symbols in your logic class: for conjunction, & or \cdot , for negation, \sim or $-$, for the conditional, \rightarrow , for the biconditional, \leftrightarrow . If you want to use the symbols with which you're familiar in this class, we won't complain, but be consistent. You might not have seen \perp ("bottom" or "das Absurde"): we will explain its meaning shortly.

Note: $(p \vee q)$ is not a formula (since p and q are not capital letters). It is, rather, a formula *schema*: a pattern that many different formulas can "fit" or "instantiate." p and q are not themselves propositional constants; they serve to mark places where an arbitrary formula may be inserted into a *schema*. Here are some instances of the schema:

- $((A \wedge B) \vee \neg A)$
- $((A \vee B) \vee (A \vee C))$

In the first case, we substituted the formula $(A \wedge B)$ for the letter p in the schema, and we substituted the formula $\neg A$ for q . In the second case, we substituted $(A \vee C)$ for q . Note that the following formulas are *not* instances of $(p \vee q)$:

- $(A \wedge (B \vee C))$
- $\neg(A \vee B)$

Convention for parentheses: The parentheses can get a bit bothersome, so we will adopt the following conventions:

- Outer parentheses may be dropped: so, for example, $A \vee B$ is an abbreviation for $(A \vee B)$.
- We will consider $(p \vee q \vee r)$ as an abbreviation for $((p \vee q) \vee r)$, and $(p \wedge q \wedge r)$ as an abbreviation for $((p \wedge q) \wedge r)$

2 Semantics

Logicians don't normally concern themselves much with *truth* simpliciter. Instead, they use a relativized notion of truth: *truth in a model*. You may not be familiar with this terminology, but you certainly understand the idea of *truth in a row of a truth table*, and in (classical) propositional logic, that is basically what truth in a model amounts to.

A model is something that provides enough information to determine truth values for *all* of the formulas in a language. How much information is required depends on the language. In the simple propositional language we're considering, we have a very limited vocabulary—propositional constants and a few truth-functional connectives—and that allows us to use very simple models. When we add quantifiers, and, later, modal operators, we will need more complex models.

What does it mean to say that a connective is *truth-functional*? It means that the only information we need to determine the truth value of a compound formula formed with one of these connectives is the truth values of the formulas it connects. Thus, for example, all we need to know to determine the truth value of $\neg B \wedge C$ are the truth values of $\neg B$ and C . And all we need to know to determine the truth value of $\neg B$ is the truth value of B . No further information about the meaning of B is needed.

Because all of our connectives are truth functional, once the truth values of the propositional constants are fixed, the truth values of *all* the formulas in the language are fixed as a result. Because of this, a *model* for classical propositional logic is just an assignment of truth values (True or False) to each propositional constant.

Although there are infinitely many propositional constants, usually we only need to concern ourselves with a few of them—those that occur in the arguments we're analyzing. Suppose the formulas we're looking at contain the constants A , B , and C . Then we can describe two different models ($\mathcal{M}1$ and $\mathcal{M}2$) by describing the truth values they give to these constants:

- $\mathcal{M}1(A) = \text{True}$, $\mathcal{M}1(B) = \text{False}$, $\mathcal{M}1(C) = \text{False}$.
- $\mathcal{M}2(A) = \text{False}$, $\mathcal{M}2(B) = \text{False}$, $\mathcal{M}2(C) = \text{True}$.

This notation is a bit tedious, though. We can present the same information in tabular form:

	A	B	C
$\mathcal{M}1$	T	F	F
$\mathcal{M}2$	F	F	T

You can see that a model is basically a row of a truth table.¹

Why are logicians interested in *truth in a model*? Because all of the fundamental semantic logical relations are defined in terms of it:

- An argument is *valid* iff there is no model in which all of its premises are True and its conclusion False. In this case the conclusion is said to be a *consequence* of the premises.
- A formula p *implies* another formula q iff there is no model in which p is True and q is False.
- Two formulas are *equivalent* iff they have the same truth value in every model.
- A set of formulas is *consistent* iff there is a model in which all are True.

¹“Basically,” because in fact a row of a truth table represents infinitely many models that agree on their assignments to the propositional constants represented in the table, but disagree on their assignments to propositional constants not listed. We can safely ignore this subtlety for most purposes.

- A formula p is a *logical truth* if it is True in every model, a *logical contradiction* or *logical falsehood* if it is False in every model, and *logically contingent* if it is neither a tautology nor a contradiction.

Sometimes the terms defined above are qualified to indicate the *kind* of models we are considering. For example, when we are considering only models of classical propositional logic, where all the connectives are truth-functional, we can talk of “truth-functional validity,” “truth-functional equivalence,” and so on, to make that clear. The term *tautology* is sometimes used for truth-functional logical truth.

As we’ve seen, in classical propositional logic, a model is just a row of a truth table. So, in classical propositional logic, a tautology is a formula that is true on all rows of a truth table; two formulas are equivalent iff they are true on all the same rows of a truth table, and so on.

Exercises:

1. Give an example of a truth-functional connective other than the usual ones (conjunction, disjunction, negation, material conditional and biconditional). Explain what makes it truth-functional. Give an example of a non-truth-functional connective, and show that it is not truth-functional.
2. Write out truth tables for the following formulas:
 - (a) $P \vee \neg(R \equiv S)$
 - (b) $Q \vee (\neg(\neg Q \wedge \neg R))$
3. What does it mean to say that a formula of propositional logic is a *tautology*? A *contradiction*? *Contingent*? In which categories do the following formulas fall?
 - (a) $P \supset (\perp \supset \neg P)$
 - (b) $P \vee (Q \wedge (\neg P \vee \neg Q))$

Note: \perp is a special propositional constant (or, if you like, a 0-place connective—a connective that takes 0 formulas and yields a new formula). It is False in every model, so when you do your truth tables, you can just write F in every row under \perp .

4. Are the following formulas consistent?

$$P \supset Q, Q \supset S, \neg S \supset \neg P$$

5. What does it mean to say that two formulas are *logically equivalent*? Give an (interesting) example of two logically equivalent formulas of propositional logic.
6. Does $P \supset (Q \wedge \neg Q)$ truth-functionally imply $\neg P \vee R$? Does $P \supset (Q \supset R)$ truth-functionally imply $R \supset (\neg Q \supset \neg P)$?

3 Proofs

There are many different proof systems for propositional logic. You probably learned some kind of *natural deduction* system. I favor natural deduction systems that keep track of undischarged assumptions using vertical lines, rather than numbers. This kind of system is sometimes called a “Fitch-style system,” after F. B Fitch, who invented it in his *Symbolic Logic*, 1952. I learned it from Fitch’s student Nuel Belnap, and my presentation here draws on his unpublished manuscript *Notes on the Art of Logic*.² However, the system we’ll use here comes from Jon Barwise and John Etchemendy, *Language, Proof, and Logic* (CSLI, 1999).

3.1 Structural Rules

“Structural rules” are so called because they do not involve any specific connectives or quantifiers. Instead, they have to do with structural aspects of proofs.

3.1.1 Hypothesis

Any formula may be written down at any time, above a horizontal line. The justification may be written “hyp,” or the justification may simply be omitted, since it is clear from the horizontal line. Alternatively, several formulas may be simultaneously hypothesized, one per line, with a horizontal line below them.

Each hypothesis begins a subproof, which we signify by a vertical line to the left of the formulas. Subsequent steps in the subproof are considered to be proved “under” the hypothesis (or hypotheses), not proved outright; that is, they are asserted as true under the supposition that the hypothesis is true, not as categorically true.

Example:

1		$S \wedge T$	hyp.	
2		Q	hyp.	
3		\vdots		
4		\vdots		
5			R	hyp.
6			\vdots	
7			\vdots	
8			\vdots	
9			P	hyp.
10			\vdots	
11			\vdots	
12			\vdots	

(1)

²<http://www.pitt.edu/~belnap/nal.pdf>. Highly recommended!

3.1.2 Availability

A formula p may be used as a premise for an inference at a particular point x in a proof just in case

- p occurs above x , and
- p occurs either
 - in the same subproof as x , or
 - in a subproof that “contains” the subproof to which x belongs.

Pictorially: you may use as premises formulas that occur *above* your present position in the proof and either on the same vertical line or on a vertical line to its *left*. For example, in proof (1) above, at line 11 you could use formulas in lines 1-4 and 8-10, but not formulas in lines 5-7 (since they belong to a subproof that does not contain the subproof to which 11 belongs).

3.2 Rules for propositional connectives

3.2.1 Tautological Consequence (*Taut Con*)

If B is a truth-functional consequence of A_1, A_2, \dots, A_n , and A_1, A_2, \dots, A_n are all available for use as premises at the current line in the proof, then you may write down B with justification “Taut Con” (citing the lines containing $A_1 \dots A_n$). It is polite to use “Taut Con” only for *obvious* implications, but technically, any implication that is checkable with a truth table will do.

Example:

1	$\neg R \wedge (S \supset R)$	
2	$S \supset R$	Taut Con 1
3	$\neg R$	Taut Con 1
4	$\neg S$	Taut Con 2, 3

(2)

Notes:

1. It would have been legal to get line 4 directly from line 1 by Taut Con. But it is often helpful to break up reasoning into bite-sized chunks.
2. When “standard” tautological eliminations are used, it is often helpful to provide descriptive names for them, in addition to “Taut Con.” For example, line 2 might be annotated “Taut Con (\wedge Elim), 1,” and line 4 might be annotated “Taut Con (MT), 2, 3.” (“MT” for “Modus tollens.”) But this is not required: “Taut Con” by itself is the official justification.

Note that if a formula is a tautology, it may be put down on any line of a proof with the justification “Taut Con,” and no line number. This is because a tautology is a truth-functional consequence of every set of formulas, including the empty set! Alternatively, you may write “Tautology.”

3.2.2 Conditional proof (\supset Intro)

If a proof contains a subproof with a single hypothesis p and last line q , you may close the subproof and write, as the very next line, the conditional $p \supset q$, with the justification “ \supset Intro” (citing the lines of the subproof).

Example:

1	$\neg P \wedge (P \vee R)$		
2	$\neg P$	Taut Con 1	
3	$P \vee R$	Taut Con 1	(3)
4	R	Taut Con 2, 3	
5	$(\neg P \wedge (P \vee R)) \supset R$	\supset Intro 1-4	

Note that the vertical line indicating the subproof ends just before the line containing the conditional conclusion (5). The hypothesis has been “discharged,” and the conditional is no longer being asserted merely “under the hypothesis” stated in line 1.

Be careful to add parentheses around the antecedent and consequent of the conditional when needed to avoid ambiguity.

3.2.3 Biconditional proof (\equiv Intro)

You prove a biconditional by combining two conditional proof subproofs, one in each direction:

1	p		
2	\vdots		
3	q		
4	q		(4)
5	\vdots		
6	p		
7	$p \equiv q$	\equiv Intro, 1-6	

3.2.4 \perp Intro and Elim

If a formula p and its negation $\neg p$ are both available for use as premises, you may write down \perp with justification “ \perp Intro.”

If \perp is available as a premise, you may write down any formula, with justification “ \perp Elim.”

The basic idea: “the absurd” proves anything (*why?*), and can be derived directly from any pair of explicitly contradictory formulas.

3.2.5 Reductio ad absurdum (\neg Intro)

If a proof contains a subproof with hypothesis p and last line \perp , you may close off the subproof and write, as the very next line, $\neg p$, with the justification “ \neg Intro” (citing the lines of the subproof).

Example:

1	Q	\dots	
2	$\neg(P \vee Q)$		
3	$\neg P \wedge \neg Q$	Taut con (De Morgan's) 2	
4	$\neg Q$	Taut Con 3	(5)
5	\perp	Taut Con 1, 4	
6	$\neg\neg(P \vee Q)$	\neg Intro 2-5	
7	$P \vee Q$	Taut Con 6	

Notes:

1. We could have gotten line 5 directly from 1 and 2 by Taut Con. But it doesn't hurt to break down the steps.
2. We can't get line 7 directly by \neg Intro, because it is not the negation of the hypothesis (though it is *equivalent* to the negation of the hypothesis). The extra step of double-negation elimination is needed.

3.2.6 Dilemma (\vee Elim)

Like the \equiv Intro rule, this involves two subproofs.

1	$p \vee q$		
2	p		
3	\vdots		
4	r		
5	q		
6	\vdots		
7	r		
8	r	\vee Elim, 1-7	(6)

4 The relation of semantics and proofs

Once we have a semantics and a proof system for our logic, we can ask questions about how they are related. Ideally, we'd like to have the following two properties:

Soundness If q can be proved from hypotheses p_1, \dots, p_n in our proof system, then q is a logical consequence of p_1, \dots, p_n ; that is, there is no model on which p_1, \dots, p_n are true and q false.

Completeness If q is a logical consequence of p_1, \dots, p_n , then it is possible to prove q from hypotheses p_1, \dots, p_n in our system.

In fact, our proof system does have both these properties relative to our semantics. But this is not just obvious. It is something that has to be proved. (If you take Philosophy 140A, you can find out how this is done.)

Exercises:

1. I've just completed a correct deduction of $\neg p$ from q in a sound deduction system. Can I conclude that q does not truth-functionally imply p ? Why or why not?
2. For each of the following arguments, either show that it is valid by giving a proof, or show that it is invalid by describing a model on which the premises are true and the conclusion false:
 - (a) $A, B \supset (A \supset B) / \therefore B$
 - (b) $A \equiv (B \vee C), A \vee B, A \vee C / \therefore A$
 - (c) $A \vee (B \supset C), B / \therefore A \vee C$

Your proofs should make use of subproofs. Don't just give a one-line proof with justification "Taut Con."