PRINCIPLES OF ANALYSIS
TOPIC 1: SYMBOLIC LOGIC

PAUL L. BAILEY

1. Propositions

A *proposition* is a statement which is either true or false, although we may not know which. Propositions are denoted by lowercase letters such as p, q or r. The truth or falsity of the proposition is called its *truth value*, and the two possible truth values are labeled **T** for TRUE and **F** for FALSE. The truth value of the proposition p is denoted $\mathbf{V}(p)$.

For example, the statement "The sun rises in the east" is a proposition, and if we wish to label this statement p, we write

$$p = \text{“The sun rises in the east”}.$$

Similarly, we may write

$$q = \text{“The sun rises in the west”}.$$

In this case, $\mathbf{V}(p) = \mathbf{T}$ and $\mathbf{V}(q) = \mathbf{F}$.

2. Logical Operators

Propositions may be modified and combined by the use of *logical operators*, which take one or more propositions and create a new one which has its own truth value. The resultant truth value is uniquely determined by the proposition(s) operated upon and the operator(s) used. Operators which accept one input are called *unary* operators, and operators which accept two inputs are called *binary* operators.

The behavior of each logical operator is determined by a *truth table*. The truth table lists all possible combinations of the truth values of the inputs, and states the operator's output for each combination of inputs.

The simplest useful logical operator is the *negation* operator NOT ($\neg$), which operates on a single proposition and reverses its truth value. Thus

$$\neg(\text{“Pigs are mammals”}) = \text{“Pigs are not mammals”}.$$

Date: August 24, 2005.

The action that NOT has on the truth value of a proposition is defined by its truth table, which lists the possible truth values of a proposition p side by side with the truth value of $\neg p$:

p	$\neg p$
T	**F**
F	**T**

Table 1. NOT Truth Table

Assertion 1. *If p is any proposition, then*

$$\mathbf{V}(\neg(\neg p)) = \mathbf{V}(p)$$

.

Proof. If p is TRUE, then $\neg p$ is FALSE, and so $\neg(\neg p)$ is TRUE. If p is FALSE, then $\neg p$ is TRUE, and so $\neg(\neg p))$ is FALSE. □

The next logical operator we consider is the *conjunction* operator AND ($\wedge$). The proposition $p \wedge q$ is true only when both p and q are true propositions. For example, if $p =$ "Pigs are mammals" and $q =$ "Pigs fly", then $p \wedge q$ may be interpreted as "Pigs are flying mammals". The AND operator is defined by a truth table which lists all possible combinations of the truth values of p and q:

p	q	$p \wedge q$
T	**T**	**T**
T	**F**	**F**
F	**T**	**F**
F	**F**	**F**

Table 2. AND Truth Table

The *disjunction* operator OR ($\vee$) returns a value of TRUE whenever either proposition it operates upon is true, and therefore is defined by:

p	q	$p \vee q$
T	**T**	**T**
T	**F**	**T**
F	**T**	**T**
F	**F**	**F**

Table 3. OR Truth Table

Thus if let p and q be as above and we assume that pigs are mammals who cannot fly, we have $\mathbf{V}(p) = \mathbf{T}$, $\mathbf{V}(q) = \mathbf{F}$, $\mathbf{V}(p \wedge q) = \mathbf{F}$ and $\mathbf{V}(p \vee q) = \mathbf{T}$.

At this point we adopt the convention that the NOT operator takes "binds tighter" than any other operator, that is, it takes precedence in the order of operations and applies only to the object on its immediate right. Thus $\neg p \wedge q$ means $(\neg p) \wedge q$ as opposed to $\neg(p \wedge q)$. We are now ready for our first theorem.

Theorem 1. *(DeMorgan's Laws) For any two propositions p and q we have*

(1) $\mathbf{V}(\neg(p \vee q)) = \mathbf{V}(\neg p \wedge \neg q)$;
(2) $\mathbf{V}(\neg(p \wedge q)) = \mathbf{V}(\neg p \vee \neg q)$.

Proof. The proofs of these assertions are truth tables in which each step is expanded, and the columns corresponding to either side of the equalities above are compared.

p	q	$p \vee q$	$\neg(p \vee q)$	$\neg p$	$\neg q$	$\neg p \wedge \neg q$
T	**T**	**T**	**F**	**F**	**F**	**F**
T	**F**	**T**	**F**	**F**	**T**	**F**
F	**T**	**T**	**F**	**T**	**F**	**F**
F	**F**	**F**	**T**	**T**	**T**	**T**

p	q	$p \wedge q$	$\neg(p \wedge q)$	$\neg p$	$\neg q$	$\neg p \vee \neg q$
T	**T**	**T**	**F**	**F**	**F**	**F**
T	**F**	**F**	**T**	**F**	**T**	**T**
F	**T**	**F**	**T**	**T**	**F**	**T**
F	**F**	**F**	**T**	**T**	**T**	**T**

□

If propositions are linked together to form new propositions via logical operators, the result may be called a *composite* proposition. Propositions which are not presented as composites are known as *atomic* propositions, or *atoms.* It is critical to realize that the propositional calculus we are developing cannot tell us anything about the truth or falsity of atoms. However, if we know the truth value of atoms prior to applying the propositional calculus to some composite of them, it will tell us the truth value of that composite.

The proof of DeMorgan's Laws points out that even complicated composites have corresponding truth tables which relate the possible truth values of potentially unknown propositions to the truth value of the composite. In particular, suppose we do not know the truth values of p and q, and we let $r = \neg(p \wedge q)$ and $s = \neg p \vee \neg q$. Then $\mathbf{V}(r) = \mathbf{V}(s)$ regardless of the meaning of p and q.

Corollary 1. *The disjunction operator OR may be defined in terms of the negation operator NOT and the conjunction operator AND as*

$$\mathbf{V}(a \vee b) = \mathbf{V}(\neg(\neg a \wedge \neg b)).$$

Proof. Apply Assertion 1 to DeMorgan's First Law (take the NOT of both sides). □

We may think of the NOT operator as distributing into the AND operator, but when it does so it changes AND to OR. An analogous statement applies to the OR operator. However, we do have a actual distributivity of AND over OR and of OR over AND.

Theorem 2. *(Distributive Laws) For any two propositions p and q we have*

(1) $\mathbf{V}((p \vee q) \wedge r) = \mathbf{V}((p \wedge r) \vee (q \wedge r))$;
(2) $\mathbf{V}((p \wedge q) \vee r) = \mathbf{V}((p \vee r) \wedge (q \vee r))$.

Proof. The tables tell the story.

p	q	r	$p \vee q$	$(p \vee q) \wedge r$	$p \wedge r$	$q \wedge r$	$(p \wedge r) \vee (q \wedge r)$
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**
T	**T**	**F**	**T**	**F**	**F**	**F**	**F**
T	**F**	**T**	**T**	**T**	**T**	**F**	**T**
T	**F**	**F**	**T**	**F**	**F**	**F**	**F**
F	**T**	**T**	**T**	**T**	**F**	**T**	**T**
F	**T**	**F**	**T**	**F**	**F**	**F**	**F**
F	**F**	**T**	**F**	**F**	**F**	**F**	**F**
F	**F**	**F**	**F**	**F**	**F**	**F**	**F**

p	q	r	$p \wedge q$	$(p \wedge q) \vee r$	$p \vee r$	$q \vee r$	$(p \vee r) \wedge (q \vee r)$
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**
T	**T**	**F**	**T**	**T**	**T**	**T**	**T**
T	**F**	**T**	**F**	**T**	**T**	**T**	**T**
T	**F**	**F**	**F**	**F**	**T**	**F**	**F**
F	**T**	**T**	**F**	**T**	**T**	**T**	**T**
F	**T**	**F**	**F**	**F**	**F**	**T**	**F**
F	**F**	**T**	**F**	**T**	**T**	**T**	**T**
F	**F**	**F**	**F**	**F**	**F**	**F**	**F**

□

Intuitively we realize that AND and OR are *commutative* operators, which is to say that $p \wedge q$ means the same thing as $q \wedge p$ and $p \vee q$ is just another way of saying $q \vee p$. Thus we are content when we notice that our truth tables agree. It is also easily verified that AND and OR are *associative* operators, and we leave it to the reader to verify this.

Assertion 2. *(Commutativity Laws) For any two propositions p and q we have*

(1) $\mathbf{V}(p \wedge q) = \mathbf{V}(q \wedge p)$;
(2) $\mathbf{V}(p \vee q) = \mathbf{V}(q \vee p)$.

Assertion 3. *(Associativity Laws) For any propositions p, q, and r we have*

(1) $\mathbf{V}((p \wedge q) \wedge r) = \mathbf{V}(p \wedge (q \wedge r))$;
(2) $\mathbf{V}((p \vee q) \vee r) = \mathbf{V}(p \vee (q \vee r))$.

Commutativity and associativity do not hold for all of the commonly used logical operators. This brings us to the *implication* operator IMP ($\Rightarrow$), where we read $p \Rightarrow q$ as "p implies q" or as "if p, then q". We have a name for the components of an implication: p is called the *hypothesis* and q is called the *conclusion*. One may be surprised by the truth table of this logical operator the first time it is encountered:

p	q	$p \Rightarrow q$
T	**T**	**T**
T	**F**	**F**
F	**T**	**T**
F	**F**	**T**

Table 4. IMP Truth Table

A false proposition implies anything one wishes it to imply. Thus the proposition "If pigs fly, then the earth if flat" is true whether or not the earth is indeed flat. Just to get our feet wet with the implication operator, we assert the following, which may be verified directly from the truth tables.

Assertion 4. *If p and q are propositions, then*

(1) $p \Rightarrow (p \vee q)$;
(2) $(p \wedge q) \Rightarrow p$.

Theorem 3. *The implication operator IMP may be built from the negation operator NOT and the conjunction operator AND operators since*

$$\mathbf{V}(p \Rightarrow q) = \mathbf{V}(\neg(p \wedge \neg q)).$$

At this point you may be asking why we chose for $p \Rightarrow q$ to be true even when p and q are both false. The others choices in the truth table for implication are easily justified by common sense, but why this one? The answer lies in the truth table for the equivalence operator and the theorem which follows it, a theorem which we very much want to be true and which depends on this choice.

The *equivalence* operator IFF ($\Leftrightarrow$) signifies logical equivalence, so that $p \Leftrightarrow q$ is read "p is logically equivalent to q" or "p if and only if q". This is the operator that answers the question "do p and q have the same truth value?"

p	q	$p \Leftrightarrow q$
T	**T**	**T**
T	**F**	**F**
F	**T**	**F**
F	**F**	**T**

Table 5. IFF Truth Table

The following theorem justifies our double sided arrow notation.

Theorem 4. *If p and q are propositions, then*

$$\mathbf{V}((p \Rightarrow q) \wedge (p \Rightarrow q)) = \mathbf{V}(p \Leftrightarrow q).$$

Proof. We have a proof by truth table.

p	q	$p \Rightarrow q$	$q \Rightarrow p$	$(p \Rightarrow q) \wedge (q \Rightarrow p)$	$p \Leftrightarrow q$
T	**T**	**T**	**T**	**T**	**T**
T	**F**	**F**	**T**	**F**	**F**
F	**T**	**T**	**F**	**F**	**F**
F	**F**	**T**	**T**	**T**	**T**

□

Theorem 5. *The equivalence operator IFF may be constructed from the negation operator NOT and the conjunction operator AND because*

$$\mathbf{V}(\neg(p \wedge \neg q) \wedge \neg(\neg p \wedge q)) = \mathbf{V}(p \Leftrightarrow q).$$

At this point we may abandon our $\mathbf{V}(p)$ notation in preference to usage of the IFF operator, for it is clear that for any two propositions p and q, then $\mathbf{V}(p) = \mathbf{V}(q)$ is the logical equivalent of $p \Leftrightarrow q$. For example, the above claim could be written

$$\mathbf{V}((\neg(p \wedge \neg q) \wedge \neg(\neg p \wedge q)) \Leftrightarrow (p \Leftrightarrow q)) = \mathbf{T},$$

or simply

$$(\neg(p \wedge \neg q) \wedge \neg(\neg p \wedge q)) \Leftrightarrow (p \Leftrightarrow q),$$

since asserting the above is taken to mean asserting that it is true.

3. Tautologies and Contradictions

In general, we need to know the truth value of the atomic components of a composite proposition in order to determine the truth value of the composite. However, this is not always the case. If a given proposition is always true regardless of the truth values of its atomic components, it is called a *tautology.* On the other hand, if a proposition is always false it is called a *contradiction.* Tautologies and contradictions are called *independent* of the truth values of the component atoms. A proposition which is neither a tautology nor a contradiction is called a *dependent*, or *indeterminate* proposition.

Examples of tautologies:

(1) $p \vee \neg p$
(2) $\neg(p \wedge \neg p)$
(3) $p \Leftrightarrow \neg(\neg p)$
(4) $\neg(p \vee q) \Rightarrow (p \Rightarrow q)$
(5) Demorgan's Laws
(6) Distributive Laws

Any two tautologies may be combined via the AND operator to form another tautology. Indeed, the tautology

$$(p \vee \neg p) \wedge \neg(p \wedge \neg p),$$

which states that either p is true or $\neg p$ is true, but not both, is often considered the basis of Western logic. Notice that the "but not both" part may be derived from the $p \vee \neg p$ part by an application of DeMorgan's Law.

Examples of contradictions:

(1) $p \wedge \neg p$
(2) $p \Rightarrow \neg p$
(3) $(p \Leftrightarrow (p \wedge q)) \wedge (p \Rightarrow q)$

Similarly, any two contradictions may be combined via the OR operator to form another contradiction (they may also be combined via the AND operator to form another contradiction, but this is a weaker statement).

Examples of indeterminate propositions:

(1) $(p \Rightarrow q) \Leftrightarrow (p \wedge q)$
(2) $(p \vee \neg q) \Rightarrow (p \vee \neg p)$
(3) $p \Rightarrow q$

In a certain sense, mathematics is the process of discovering tautologies. However, the superstructure of most theorems is of the indeterminate form $p \Rightarrow q$. Why, then, is it difficult to prove theorems? It may seem that one simply needs to determine the truth values of p and q and verify the truth or falsity of the theorem with a glance at the truth table for implication. This is far from the case; an implication is a description of the relationship between p and q, and not of their individual truth values. In fact, proving an implication involves verifying that all four rows of the truth table for implication are satisfied (although such proofs rarely take this explicit form).

Now we turn to a pair of constructions which are critically important for aspiring mathematicians to grasp. Suppose that p and q are propositions, and consider the implication $p \Rightarrow q$. The *converse* of this implication is the proposition $q \Rightarrow p$, whereas its *contrapositive* is the proposition $\neg q \Rightarrow \neg p$.

Assertion 5. *The contrapositive of an implication is logically equivalent to it. The converse of an implication is logically independent of it.*

Proof. To explore the logical relations between any two propositions a and b, we construct the truth table of $a \Leftrightarrow b$. If this truth table contains nothing but **T**'s in the last column, then a and b are logically equivalent. If this truth table contains nothing but **F**'s in the last column, then a and b are logically incompatible. If this truth table contains some **T**'s and some **F**'s in its last column, then a and b are logically independent. We leave it as an exercise to determine what a and b should be in these cases and to complete the proof. □

An example is in order. Let p be the proposition “The egg falls fifty feet onto cement” and q be the proposition “The egg breaks”. Additionally, we assume that when an egg falls fifty feet onto cement, then it breaks, so that we are assuming that $p \Rightarrow q$ is true. Now it is clear that if the egg is not broken, it could not have fallen fifty feet onto cement. This is nothing more than the claim $\neg q \Rightarrow \neg p$. On the other hand, it is possible to break an egg without dropping it fifty feet onto cement; just because it is broken, we may not accurately conclude that it did drop fifty feet onto cement. So the converse $q \Rightarrow p$ is not necessarily true.

It is intuitively clear that the converse of an implication is not logically equivalent to the implication, and yet when immersed in the abstract world of mathematics, surrounded by definitions and related ideas which have not previously been contemplated, the distinction between an implication and its converse may seem to blur. Thus it is a good idea to keep in mind “the converse is not necessarily true” (even when the implication is).

On the other hand, many proofs depend on the contrapositive. It is often easier to prove that $\neg q \Rightarrow \neg p$ than $p \Rightarrow q$; but if we can prove that $\neg q \Rightarrow \neg p$, we get $p \Rightarrow q$ for free.

A related idea is that of proof by contradiction. Here we wish to prove some proposition a, where a may or may not be in the form of an implication. The roundabout method of proof by contradiction assumes that $\neg a$ is true, and arrives at a conclusion which is a proposition known to always be false, in other words, a contradiction. Thus the assumption that led to the contradiction ($\neg a$) must be false, proving that a is true. This technique is invaluable in group theory and topology.

Often one finds proofs that masquerade as proofs by contradiction but are actually proofs by contrapositive. That is, one wishes to prove that $p \Rightarrow q$, and so assumes that $p \wedge \neg q$ is true, and arrives at a contradiction, without ever using the assumption p. This is not the preferred method.

4. Generation of Operators

In this section we introduce primitive logical operators which do not arise in ordinary language but which, nonetheless, arise from definitional truth tables which differ from those we have already encountered. These are XOR, NOR, and NAND.

The *exclusion* operator XOR ($\updownarrow$) stands for exclusive OR and means a or b, but not both.

a	b	$a \updownarrow b$
T	**T**	**F**
T	**F**	**T**
F	**T**	**T**
F	**F**	**F**

Table 6. XOR Truth Table

Assertion 6. *The XOR operator is the negation of IFF, i.e.,*

$$(a \updownarrow b) \Leftrightarrow \neg(a \Leftrightarrow b).$$

The *alternate denial* operator NOR ($\uparrow$) means "neither a nor b".

a	b	$a \uparrow b$
T	**T**	**F**
T	**F**	**F**
F	**T**	**F**
F	**F**	**T**

TABLE 7. NOR Truth Table

Assertion 7. *The NOR operator is the negation of OR, i.e.,*

$$(a \uparrow b) \Leftrightarrow \neg(a \vee b).$$

The *joint denial* operator NAND ($\downarrow$) means "possibly a and possibly b, but not both".

a	b	$a \downarrow b$
T	**T**	**F**
T	**F**	**T**
F	**T**	**T**
F	**F**	**T**

TABLE 8. NAND Truth Table

Assertion 8. *The NAND operator is the negation of AND, i.e.,*

$$(a \downarrow b) \Leftrightarrow \neg(a \wedge b).$$

A collection of operators *generates* another operator if the truth table of generated operator can be derived through a combination of the generators. For example, we have already seen that NOT and AND together generate OR, IMP, and IFF. Since XOR is NOT IFF, NOR is NOT OR, and NAND is NOT AND, we can see that NOT and AND generate XOR, NOR, and NAND.

Theorem 6. *The operators NOT, AND, OR, IMP, IFF, XOR, and NAND may be derived from NOR.*

Proof. It suffices to show that NOT and AND may be written in terms of NOR. The definition of NOR and DeMorgan's Law gives us that

(1) $\neg a \Leftrightarrow (a \uparrow a)$;
(2) $(a \wedge b) \Leftrightarrow (\neg a \uparrow \neg b)$.

□

Theorem 7. *The operators NOT, AND, OR, IMP, IFF, XOR, and NOR may be derived from NAND.*

Proof. It suffices to show that NOT and AND may be written in terms of NAND. The definition of NAND and a glance at the truth tables gives us that

(1) $\neg a \Leftrightarrow (a \downarrow a)$
(2) $(a \wedge b) \Leftrightarrow \neg(a \downarrow b)$

□

There are four possible logical operators of a single proposition, and we have only discussed the identity operator ($\mathbf{V}$) and NOT. There are also the constant operators whose value is always **T** or **F**. Notice that a constant operator cannot be generated from NOT because NOT NOT is the identity, NOT NOT NOT is NOT, etc. We use this fact in our final theorem.

Theorem 8. *The operators NOR and NAND are the only binary operators which are sufficient by themselves to generate NOT, AND, OR, IMP, IFF, XOR, NOR, and NAND.*

Proof. In order for a generic binary operator GEN ($\pitchfork$) to generate NOT, $a \pitchfork b$ must be false when both a and b are true, for otherwise we can never achieve anything but true in the first row of a truth table of a composite proposition whose only operator is GEN. Similarly, $a \pitchfork b$ must be true whenever both a and b are false. Thus we have a partial truth table for GEN.

p	q	$p \pitchfork q$
T	**T**	**F**
T	**F**	$\mathbf{V}_1$
F	**T**	$\mathbf{V}_2$
F	**F**	**T**

Now suppose that GEN is not a commutative operator. If $\mathbf{V}_1 = \mathbf{T}$ and $\mathbf{V}_2 = \mathbf{F}$, then $(p \pitchfork q) \Leftrightarrow \neg(q)$ is a tautology, and if $\mathbf{V}_1 = \mathbf{F}$ and $\mathbf{V}_2 = \mathbf{T}$, then $(p \pitchfork q) \Leftrightarrow \neg(p)$ is a tautology. In either case, GEN may be constructed from NOT. However, NOT cannot generate a constant operator of a single atom such as $p \wedge \neg p$, which is always false, and thus NOT cannot generate AND.

Thus for GEN to generate the other logical operators, it must be commutative so that $\mathbf{V}_1 = \mathbf{V}_2 = \mathbf{V}$. If $\mathbf{V} = \mathbf{T}$, then GEN is NAND, and if $\mathbf{V} = \mathbf{F}$, then GEN is NOR. □

There are sixteen possible truth tables resulting from combinations of two propositions, and we have only mentioned seven of them. The reader is welcomed to explore the possibilities inherent in the others.

5. Exercises

Exercise 1. Determine the truth table of the following composite propositions and state whether they are tautologies, contradictions, or indeterminate.

(a) $(p \vee q) \Rightarrow (p \wedge q)$
(b) $(p \wedge q) \vee (p \Rightarrow q)$
(c) $(p \Rightarrow q) \Rightarrow p$
(d) $p \Rightarrow (q \Rightarrow p)$
(e) $(p \Rightarrow q) \Rightarrow q$
(f) $p \Rightarrow (q \Rightarrow p)$
(g) $(p \Rightarrow q) \Rightarrow r$
(h) $p \Rightarrow (q \Rightarrow r)$
(i) $((p \Rightarrow q) \wedge (q \Rightarrow r)) \Rightarrow (p \Rightarrow r)$
(j) $(p \wedge q) \Leftrightarrow (p \updownarrow q)$
(k) $(p \downarrow q) \Rightarrow (p \vee q)$

Exercise 2. Complete the proof of Assertion 5.

Exercise 3. Write a logically equivalent statement using NOT, AND, and OR.

(a) $\neg(p \Rightarrow q)$
(b) $(p \Rightarrow q) \Rightarrow r$

Exercise 4. Use truth tables to prove the following assertions.

(a) $(a \updownarrow b) \Leftrightarrow \neg(a \Leftrightarrow b)$
(b) $(a \uparrow b) \Leftrightarrow \neg(a \vee b)$
(c) $(a \downarrow b) \Leftrightarrow \neg(a \wedge b)$

Exercise 5. Show that the logical operators NOT and OR are sufficient to generate AND, IMP, IFF, XOR, NOR, and NAND.

Exercise 6. Develop the truth tables for logical operators of one proposition other than NOT. You should get three of these, and you will see that they may reasonably be called identity, constant truth, and constant falsehood.

Exercise 7. Develop the truth tables for logical operators of two propositions other than AND, OR, IMP, IFF, XOR, NOR, and NAND. You should get nine of these. Give these new operators names. Relate them to the operators of one proposition identity, constant truth, constant falsehood, and negation. Relate them to the operators of two propositions AND, OR, IMP, IFF, XOR, NOR, and NAND.

Department of Mathematics and CSci, Southern Arkansas University
E-mail address: `plbailey@saumag.edu`