

MANY-VALUED LOGICS

Routledge Companion to the Philosophy of Language, Article 2.6

Nicholas J.J. Smith

24 December 2009

1 Introduction

A many-valued (aka multiple- or multi-valued) *semantics*, in the strict sense, is one which employs more than two truth values; in the loose sense it is one which countenances more than two truth statuses. So if, for example, we say that there are only two truth values—True and False—but allow that as well as *possessing the value True* and *possessing the value False*, propositions may also have a third truth status—*possessing neither truth value*—then we have a many-valued semantics in the loose but not the strict sense. A many-valued *logic* is one which arises from a many-valued semantics and does not also arise from any two-valued semantics [Malinowski, 1993, 30]. By a ‘logic’ here we mean either a set of *tautologies*, or a *consequence relation*. We can best explain these ideas by considering the case of classical propositional logic. The language contains the usual basic symbols (propositional constants p, q, r, \dots ; connectives $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$; and parentheses) and well-formed formulas are defined in the standard way. With the language thus specified—as a set of well-formed formulas—its semantics is then given in three parts. (i) A *model* of a logical language consists in a free assignment of semantic values to basic items of the non-logical vocabulary. Here the basic items of the non-logical vocabulary are the propositional constants. The appropriate kind of semantic value for a proposition is a *truth* value, and so a model of the language consists in a free assignment of truth values to basic propositions. Two truth values are countenanced: 1 (representing truth) and 0 (representing falsity). (ii) Rules are presented which determine a truth value for *every* proposition of the language, given a model. The most common way of presenting these rules is via truth tables (Figure 1). Another way of stating such rules—which will be useful below—is first to introduce functions on the truth values themselves: a unary function \neg^* and four binary functions $\wedge^*, \vee^*, \rightarrow^*$ and \leftrightarrow^* (Figure 2). Representing the truth value of α (on a given model) as $[\alpha]$, we then specify the truth values of compound formulas as in Figure 3. Once one becomes familiar with the distinction between connectives and truth functions, it is customary to use the same symbols for both and to let context disambiguate. As it generally increases readability, I

α	β	$\neg\alpha$	$\alpha \wedge \beta$	$\alpha \vee \beta$	$\alpha \rightarrow \beta$	$\alpha \leftrightarrow \beta$
1	1	0	1	1	1	1
1	0		0	1	0	0
0	1	1	0	1	1	0
0	0		0	0	1	1

Figure 1: Classical truth tables

	\neg^*	\wedge^*	\vee^*	\rightarrow^*	\leftrightarrow^*
1 1	0	1	1	1	1
1 0		0	1	0	0
0 1	1	0	1	1	0
0 0		0	0	1	1

Figure 2: Classical truth functions

$$\begin{aligned}
[\neg\alpha] &= \neg^* [\alpha] \\
[\alpha \wedge \beta] &= [\alpha] \wedge^* [\beta] \\
[\alpha \vee \beta] &= [\alpha] \vee^* [\beta] \\
[\alpha \rightarrow \beta] &= [\alpha] \rightarrow^* [\beta] \\
[\alpha \leftrightarrow \beta] &= [\alpha] \leftrightarrow^* [\beta]
\end{aligned}$$

Figure 3: Rules for assigning truth values to compound propositions

shall mostly follow this practice below (i.e. omit the \star 's on truth functions). (iii) Definitions of *tautology* and *logical consequence* are introduced. In this case, a tautology is a proposition which gets the value 1 on every model (e.g. $p \vee \neg p$, $p \rightarrow p$), and a proposition α is a logical consequence of the set of propositions Γ (written $\Gamma \models \alpha$) if, on every model on which every proposition in Γ has the value 1, α has the value 1 (e.g. $\{p, p \rightarrow q\} \models q$, $\{p\} \models p \vee q$). Classical *logic* is then the language just introduced together with either the set of tautologies, or the consequence relation, just defined. The definition of a logic in terms of a consequence relation is more powerful, in that once we have the consequence relation, we can reconstruct the set of tautologies as the set of propositions α such that $\emptyset \models \alpha$. However sometimes we are interested only in tautologies—hence we allow that a logic may be specified just by giving a set of tautologies, without a consequence relation.

As we have just seen, a logic can be specified as the one which arises from a certain semantics. Logics can also be specified in terms of proofs. Proofs come in many different forms. Consider, for example, axiomatic proof systems. Finitely many propositions are taken as axioms. One or more rules of inference are specified: they take one or more propositions as input, and give a proposition as output. A formal proof (in a given axiomatic system) is then defined as a finite list of propositions, each of which is either an axiom, or follows from earlier propositions in the list by a rule of inference. A proposition α is said to be a theorem of the system if there is a proof whose last line is α . There are many known axiom systems which have the property that their theorems are exactly the tautologies of classical logic. Thus, classical logic is said to be finitely axiomatizable. We can also define consequence relations using axiomatic proof procedures, by allowing assumptions in proofs. So axiomatic proof systems give us an alternative way of characterizing logics.

In §2 we look at systems of many-valued semantics and their associated logics and in §3 we mention some of the uses to which these systems have been put. Overall space constraints, together with the judgement that readers coming to the existing literature from a philosophy of language background will find it harder to gain an overview of the different kinds of many-valued systems than to find information on applications of one or other of these systems to particular topics of interest, led to the decision to devote the bulk of the available space to §2.

2 Systems of Many-Valued Logic

2.1 Three Values

Suppose we take the language of classical propositional logic, and give it a semantics which countenances a third truth value—which we shall write as $*$ —as well as the classical 1 and 0. Taking a (three-valued) model to be a free assignment of one of these three values to each basic proposition, the classical truth tables will no longer determine a truth value for every proposition of the language. For example, if p has the value $*$ on a given model, then the tables do not specify a value for compound propositions (e.g. $\neg p$, $p \rightarrow q$) which have p as a component. One way to remedy this situation is to replace the classical tables with three-valued truth tables. There are many such tables (see Bolc and Borowik [1992, ch.3] for a survey); we begin with the three that have played the most prominent role in the literature. Figure 4 shows the Bochvar (aka Kleene weak) tables [Rescher, 1969, 29–30] [Kleene, 1952, 334]. The idea here is that a compound whose components

α	β	$\neg\alpha$	$\alpha \wedge \beta$	$\alpha \vee \beta$	$\alpha \rightarrow \beta$	$\alpha \leftrightarrow \beta$
1	1	0	1	1	1	1
1	*		*	*	*	*
1	0		0	1	0	0
*	1	*	*	*	*	*
*	*		*	*	*	*
*	0		*	*	*	*
0	1	1	0	1	1	0
0	*		*	*	*	*
0	0		0	0	1	1

Figure 4: Bochvar (aka Kleene weak) tables

all take classical values takes the value that the classical tables assign, while if any of its components takes the value $*$, the compound takes the value $*$. Figure 5 shows the Kleene (strong) tables [Kleene, 1952, 334]. As before, a compound whose components all take classical values takes the value that the classical tables assign. As for the case where a component has the value $*$, there are two useful ways of thinking about what the tables dictate. First, we can think of the $*$ as an *unknown classical* value. So we suppose the $*$ is a 1 and calculate the value that the classical table would give, and we suppose the $*$ is a 0 and calculate the value that the classical table would give: if we get 1 both times, then that is the value in the new table; if we

α	β	$\neg\alpha$	$\alpha \wedge \beta$	$\alpha \vee \beta$	$\alpha \rightarrow \beta$	$\alpha \leftrightarrow \beta$
1	1	0	1	1	1	1
1	*		*	1	*	*
1	0		0	1	0	0
*	1	*	*	1	1	*
*	*		*	*	*	*
*	0		0	*	*	*
0	1	1	0	1	1	0
0	*		0	*	1	*
0	0		0	0	1	1

Figure 5: Kleene (strong) tables

get 0 both times, then that is the value in the new table; and if we get 1 once and 0 once, then the value in the new table is *. (When we are trying to determine what value should be in the new table where *two* components both have the value *, we calculate all four possibilities given by replacing each * with a 1 or a 0.) Second, we can think of * as lying *between* 1 and 0 on a scale from more true to less true: * is more true than 0 but less true than 1. Then we can see the conjunction as taking the *least* true of the values of its conjuncts, the disjunction as taking the *most* true of the values of its conjuncts, and the conditional and biconditional as defined in the standard classical way from the other connectives: $\alpha \rightarrow \beta =_{df} \neg\alpha \vee \beta$ (or $\neg(\alpha \wedge \neg\beta)$) and $\alpha \leftrightarrow \beta =_{df} (\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$. Figure 6 shows the Łukasiewicz tables [Łukasiewicz and Tarski, 1930]. These are exactly like the

α	β	$\neg\alpha$	$\alpha \wedge \beta$	$\alpha \vee \beta$	$\alpha \rightarrow \beta$	$\alpha \leftrightarrow \beta$
1	1	0	1	1	1	1
1	*		*	1	*	*
1	0		0	1	0	0
*	1	*	*	1	1	*
*	*		*	*	1	1
*	0		0	*	*	*
0	1	1	0	1	1	0
0	*		0	*	1	*
0	0		0	0	1	1

Figure 6: Łukasiewicz tables

Kleene tables except for the values of $\alpha \rightarrow \beta$ and $\alpha \leftrightarrow \beta$ when α and β both have the value $*$: in the Łukasiewicz tables these values are 1, whereas in the Kleene tables they are $*$.

We come now to the third stage of presenting a semantics: defining notions of *tautology* and/or *consequence*. The standard way of defining these notions is to specify a subset of the truth values as *designated*. A tautology is then a proposition which takes a designated value on every model, and α is a logical consequence of Γ iff, on every model on which every proposition in Γ has a designated value, α has a designated value. (The earlier definitions in the classical case emerge from this template by setting 1 as the only designated value amongst the two classical values 1 and 0. Marking designated values by underlining, we represent this choice as $\{0, \underline{1}\}$.) In the three-valued case, the most obvious choice is to set 1 as the only designated value: $\{0, *, \underline{1}\}$. However it is also not unreasonable to set both 1 and $*$ as designated values: $\{0, *, \underline{1}\}$. On the former choice, a tautology is a proposition which is *always true* (and consequence is a matter of preservation of truth); on the later choice, a tautology is a proposition which is *never false* (and consequence is a matter of preservation of non-falsity). By combining different choices of designated values with different truth tables, we get different logics:

B_3 : $\{0, *, \underline{1}\}$ and Bochvar tables

B'_3 : $\{0, *, \underline{1}\}$ and Bochvar tables

K_3 : $\{0, *, \underline{1}\}$ and Kleene tables

K'_3 , aka LP [Priest, 2008, 124]: $\{0, *, \underline{1}\}$ and Kleene tables

L_3 : $\{0, *, \underline{1}\}$ and Łukasiewicz tables

L'_3 : $\{0, *, \underline{1}\}$ and Łukasiewicz tables

These logics differ more or less from classical logic. For a start, it is not hard to see that any semantics in which 0 is not a designated value and whose tables agree with the classical tables where only 1's and 0's are involved—this includes all the systems just introduced—will be such that all its tautologies are classical tautologies. Going the other way, all classical tautologies come out as tautologies of B'_3 and K'_3 (given the original stipulation that a many-valued logic is one which arises from a many-valued semantics and does not also arise from any two-valued semantics, this means that these logics—considered as sets of tautologies (the situation is different when we consider consequence relations)—are *not* many-valued logics: for while they do arise

from many-valued semantics, they also arise from the classical two-valued semantics); B_3 and K_3 , however, have no tautologies at all; L_3 has some of the classical tautologies (e.g. $p \rightarrow p$) but not all (e.g. $p \vee \neg p$ has the value $*$ when p does); L'_3 has all the tautologies of L_3 and some more besides (e.g. $p \vee \neg p$) but it still does not have all the classical tautologies (e.g. $\neg(p \rightarrow \neg p) \vee \neg(\neg p \rightarrow p)$ has the value 0 when p has the value $*$). The story is different again when it comes to consequence relations; for example, while B'_3 and K'_3 have the same tautologies as classical logic, they do not have classical consequence relations (e.g. q is a consequence of $\{p \wedge \neg p\}$ in classical logic, but not in B'_3 or K'_3 : consider a model on which p has the value $*$ —hence so does $p \wedge \neg p$ —and q has the value 0).

2.2 *Finitely Many Values*

The three-valued tables of the previous section can all be generalized to the case where we have any finite number n of truth values. It is convenient to represent these n values as fractions:

$$\frac{0}{n-1}, \frac{1}{n-1}, \dots, \frac{n-2}{n-1}, \frac{n-1}{n-1}$$

So where $n = 5$, for example, we have the following values: $0, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}, 1$. To generalize the Bochvar tables, we say that $[\alpha]$ is whatever the classical tables dictate when all α 's components have the values 0 or 1; and otherwise it is the 'middle value'. (Where the number n of values is odd, the middle value is simply $\frac{1}{2}$, i.e. half of the top value 1; where n is even, the 'middle' value is taken to be $\frac{n-2}{2(n-1)}$, i.e. half of the second-top value [Rescher, 1969, 43–4].) To generalize the Kleene tables, we use the rules in Figure 7. To generalize the

$$\begin{array}{ll} [\neg\alpha] & = 1 - [\alpha] \\ [\alpha \wedge \beta] & = \min([\alpha], [\beta]) \\ [\alpha \vee \beta] & = \max([\alpha], [\beta]) \\ [\alpha \rightarrow \beta] & = [\neg\alpha \vee \beta] \\ [\alpha \leftrightarrow \beta] & = [(\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)] \end{array}$$

Figure 7: Rules for n -valued Kleene systems

Lukasiewicz tables, we use rules exactly like those for the Kleene systems, except that we replace the rule for the conditional with the following:

$$[\alpha \rightarrow \beta] = \min(1, 1 - [\alpha] + [\beta])$$

(In the rule for the biconditional in the Łukasiewicz systems, the conditional is then this Łukasiewicz conditional, not the Kleene conditional.) The idea here is that if the consequent is at least as true as the antecedent, then the conditional is completely true, while if the antecedent is truer than the consequent—and the difference between their truth values is k —then the conditional has the value $1 - k$, i.e. it is k less than fully true. It is routine to verify that if we set $n = 3$ then the three sets of rules just given specify the three sets of tables given in the previous section (with $*$ written as $\frac{1}{2}$) and if we set $n = 2$ then these rules all specify the classical truth tables.

If we now choose a set of designated values, logics emerge. For example, taking the Łukasiewicz rules together with 1 as the only designated value yields the n -valued logics L_n . Lindenbaum showed that the set of tautologies of L_m is a subset of the set of tautologies of L_n ($m, n \geq 2$) just in case $n - 1$ is a divisor of $m - 1$ [Łukasiewicz and Tarski, 1930, 48]. So, for example, the tautologies of L_9 are a subset of those of L_5 (and likewise of L_3), because 4 (and 2) is a divisor of 8. Note that it follows that no tautologies can ever be *gained* as we increase the number of truth values. The logics L_n are all finitely axiomatizable (see e.g. Malinowski [1993, 39]), but not every finitely many-valued logic is; for example, Rescher [1969, 157–9] presents a three-valued logic which is not finitely axiomatizable (given substitution and modus ponens as rules of inference).

We turn now to a different strategy for generating n -valued systems, due to Post [1920, 1921]. Note that in the Kleene and Łukasiewicz (but not Bochvar) systems, the truth value of $\neg\alpha$ is as far below 1 as the truth value of α is above 0; that is, the distance between 0 and $[\alpha]$ is the same as the distance between $[\neg\alpha]$ and 1. This treatment of negation requires that there be a meaningful notion of *distance* between the truth values of the system. In Post's systems, by contrast, the truth values are merely *ordered*: given any two truth values, we can say which of them is the truer; but we cannot compare the *distances* between different pairs of values. We represent the truth values of the n -valued Post system as follows:

$$t_1, t_2, \dots, t_n$$

The ordering of the values is this: t_i is less true than t_j just in case $i < j$. Figure 8 shows the truth functions of this system. So the value of $\neg\alpha$ is the value immediately after the value of α (in the ordering of the truth values from least true to most true), except in the case where the value of α is the top value t_n , in which case the value of $\neg\alpha$ is the bottom value t_1 . The rule for disjunction is familiar from Kleene and Łukasiewicz: the value of the disjunction is the truer of the values of the disjuncts. The remaining

$$\begin{aligned}
\neg t_i &= \begin{cases} t_{i+1} & \text{if } i \neq n \\ t_1 & \text{if } i = n \end{cases} \\
t_i \vee t_j &= t_{\max(i,j)} \\
t_i \wedge t_j &= \neg(\neg t_i \vee \neg t_j) \\
t_i \rightarrow t_j &= \neg t_i \vee t_j \\
t_i \leftrightarrow t_j &= (t_i \rightarrow t_j) \wedge (t_j \rightarrow t_i)
\end{aligned}$$

Figure 8: Truth functions for n -valued Post systems

connectives are defined in the standard ways from \neg and \vee . Taking Post's rule and setting $n = 3$ yields a three-valued system different from any of the systems examined in §2.1; setting $n = 2$ yields classical logic.

It is a familiar fact that in classical logic, some connectives can be defined in terms of others. Similar kinds of results hold in many-valued logics. For example, Łukasiewicz took \neg and \rightarrow as primitive connectives, and defined the others in terms of them (Figure 9). Similarly, Bochvar took \neg and

$$\begin{aligned}
\alpha \vee \beta &=_{df} (\alpha \rightarrow \beta) \rightarrow \beta \\
\alpha \wedge \beta &=_{df} \neg(\neg\alpha \vee \neg\beta) \\
\alpha \leftrightarrow \beta &=_{df} (\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)
\end{aligned}$$

Figure 9: Defined connectives in Łukasiewicz systems

\wedge as primitive, and Post took \neg and \vee as primitive. It is also sometimes possible to define the connectives of one system within another system. For example, Kleene conditional is definable in terms of Kleene negation and disjunction (Figure 7); but Kleene negation and disjunction coincide with Łukasiewicz negation and disjunction; so Kleene conditional is definable in terms of Łukasiewicz negation and disjunction. In fact all the Kleene connectives are definable in terms of Łukasiewicz connectives, but not vice versa; and all the Bochvar connectives are definable in terms of Kleene (and hence Łukasiewicz) connectives, but not vice versa [Bergmann, 2008, 91–2]. A set of connectives is said to be *functionally complete* if we can define *all* possible connectives from the connectives in that set. In an n -valued system, an m -place connective has a truth table with n^m rows (one for each possible assignment of the n truth values to the m component propositions). We specify an m -place connective by putting a truth value in each row of the table. Thus there are $n^{(n^m)}$ possible m -place connectives. It is well known that in the classical (two-valued) case, all these connectives can be defined using

only the five standard connectives introduced at the outset. (Indeed, we do not need all of them: just \neg and \wedge will do, or \neg and \rightarrow , etc.) This property of functional completeness of (some subset of) the set of five standard connectives carries over to some, but not all, many-valued logics. In every n -valued Post system, the set containing \neg and \vee is functionally complete. In no n -valued Łukasiewicz system is the set of five standard connectives functionally complete—although for some (but not all) n this set is precomplete, which means that it is not functionally complete, but becomes so with the addition of *any* connective which is not already definable in terms of the standard five [Urquhart, 2001, 266–8].

In all the n -valued systems that we have considered so far, the set of truth values is linearly ordered—but it need not be so. For example, Belnap [1977] considers a system with four truth values— T , F , B and N —ordered as in Figure 10 (with $x < y$ iff one can get from x to y by following arrows). N

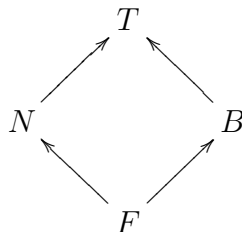


Figure 10: A lattice of four truth values

and B are distinct truth values, with neither greater than the other—so we cannot take conjunction as min and disjunction as max. However, every pair of values has a supremum (a least value that is greater than or equal to both values in the pair) and an infimum (a greatest value that is less than or equal to both values in the pair). A partially ordered structure with this property is called a lattice. In any lattice of truth values, the strategy of defining conjunction and disjunction as min and max generalizes to defining them as inf and sup. So, for example, we can take $N \wedge B = F$ and $T \wedge B = B$, and $N \vee B = T$ and $N \vee F = N$.

We turn now to a new strategy for generating finitely many-valued systems, due to Jaśkowski [1936]. Suppose we have an m -valued system X and an n -valued system Y . We can then form a new, $(m \times n)$ -valued system $X \times Y$ by taking the *product* of these two systems. The truth values of the product system will be elements of the Cartesian product $X_v \times Y_v$, where X_v (Y_v) is the set of values of X (Y); that is, they will be pairs whose first

element is a value of X and whose second element is a value of Y . Truth functions of the product system are then specified coordinatewise (Figure 11: superscripts on truth functions indicate to which system they belong). One

$$\begin{aligned}
\neg \langle x, y \rangle &= \langle \overset{X}{\neg} x, \overset{Y}{\neg} y \rangle \\
\langle x_1, y_1 \rangle \wedge \langle x_2, y_2 \rangle &= \langle x_1 \overset{X}{\wedge} x_2, y_1 \overset{Y}{\wedge} y_2 \rangle \\
\langle x_1, y_1 \rangle \vee \langle x_2, y_2 \rangle &= \langle x_1 \overset{X}{\vee} x_2, y_1 \overset{Y}{\vee} y_2 \rangle \\
\langle x_1, y_1 \rangle \rightarrow \langle x_2, y_2 \rangle &= \langle x_1 \overset{X}{\rightarrow} x_2, y_1 \overset{Y}{\rightarrow} y_2 \rangle \\
\langle x_1, y_1 \rangle \leftrightarrow \langle x_2, y_2 \rangle &= \langle x_1 \overset{X}{\leftrightarrow} x_2, y_1 \overset{Y}{\leftrightarrow} y_2 \rangle
\end{aligned}$$

Figure 11: Truth functions for product systems

system of particular interest is the product of the classical system with itself. This system has four values:

$$\langle 0, 0 \rangle, \langle 0, 1 \rangle, \langle 1, 0 \rangle, \langle 1, 1 \rangle$$

Its truth table for negation (for example) is shown in Figure 12. Note that

α	$\neg \alpha$
$\langle 0, 0 \rangle$	$\langle 1, 1 \rangle$
$\langle 0, 1 \rangle$	$\langle 1, 0 \rangle$
$\langle 1, 0 \rangle$	$\langle 0, 1 \rangle$
$\langle 1, 1 \rangle$	$\langle 0, 0 \rangle$

Figure 12: Negation in the self-product of classical logic

its truth tables for conjunction and disjunction could alternatively be arrived at by setting $\langle x_1, y_1 \rangle \leq \langle x_2, y_2 \rangle$ iff $(x_1 \leq x_2 \text{ and } y_1 \leq y_2)$, and then following the inf/sup strategy of the previous paragraph. When it comes to choosing designated for a product system, there are two obvious choices: $\langle x, y \rangle$ is designated iff x is designated in system X and/or (choose one) y is designated in system Y . In a *self*-product system $X \times X$, both choices yield the same set of tautologies—which is simply the set of tautologies of X [Rescher, 1969, 101].

2.3 Infinitely Many Values

If we wish to countenance more than finitely many truth values, there are two particularly natural options: we can take as truth values all the *rational*

numbers between 0 and 1 inclusive, or the *real* interval $[0, 1]$ (containing all the real numbers between 0 and 1 inclusive). Łukasiewicz considered both these options. The rules for assigning truth values to compound propositions stated in §2.2 carry over unchanged. If we take 1 as the only designated values, the two resulting logics are named L_{\aleph_0} and L_{\aleph_1} respectively. Interestingly, they both have the same set of tautologies [Rescher, 1969, 38–9].

A semantics which takes $[0, 1]$ as its set of truth values (and treats the connectives truth-functionally) is called a fuzzy semantics; a logic that arises from a fuzzy semantics is a fuzzy logic. Amongst philosophers, the best-known rules for assigning truth values to compound propositions in a fuzzy semantics are the Zadeh rules; the statement of these rules is exactly the same as that of the Kleene rules in Figure 7. Amongst logicians, the fuzzy systems of most interest are the t-norm fuzzy logics. Where $\overset{\star}{\wedge}$ is a binary function on $[0, 1]$ which we are going to use to define conjunction, it is natural to want $\overset{\star}{\wedge}$ to satisfy the conditions shown in Figure 13. A binary function on $[0, 1]$

$$\begin{array}{rcl}
x \overset{\star}{\wedge} y & = & y \overset{\star}{\wedge} x \\
(x \overset{\star}{\wedge} y) \overset{\star}{\wedge} z & = & x \overset{\star}{\wedge} (y \overset{\star}{\wedge} z) \\
x_1 \leq x_2 & \text{implies} & x_1 \overset{\star}{\wedge} y \leq x_2 \overset{\star}{\wedge} y \\
y_1 \leq y_2 & \text{implies} & x \overset{\star}{\wedge} y_1 \leq x \overset{\star}{\wedge} y_2 \\
1 \overset{\star}{\wedge} x & = & x \\
0 \overset{\star}{\wedge} x & = & 0
\end{array}$$

Figure 13: Conditions on t-norms

which satisfies these conditions is called a *t-norm*. If it is also continuous—in the usual sense—then it is a continuous t-norm. A t-norm fuzzy logic is determined by a choice of a continuous t-norm as the truth function for conjunction: truth functions for the other connectives are then defined in terms of this t-norm. The truth function $\overset{\star}{\rightarrow}$ for conditional is the *residuum* of the t-norm:

$$x \overset{\star}{\rightarrow} y = \max\{z : x \overset{\star}{\wedge} z \leq y\}$$

and the truth function $\overset{\star}{\neg}$ for negation is the *precomplement* of this conditional:

$$\overset{\star}{\neg} x = x \overset{\star}{\rightarrow} 0$$

Biconditional is then defined in the standard way using conditional and conjunction. Disjunction will be discussed below. The most important examples

$$\begin{array}{ll}
\text{Łukasiewicz:} & x \overset{\star}{\wedge} y = \max(0, x + y - 1) \\
\text{Gödel:} & x \overset{\star}{\wedge} y = \min(x, y) \\
\text{Product:} & x \overset{\star}{\wedge} y = x \cdot y
\end{array}$$

Figure 14: Important t-norms

of continuous t-norms are shown in Figure 14. These three are fundamental in the sense that every continuous t-norm is a combination of them [Hájek, 1998, 32]. Note that the Gödel t-norm is the min operation used to define conjunction in Zadeh/Kleene logic; it is the only *idempotent* t-norm (i.e. one which satisfies the condition $x \overset{\star}{\wedge} x = x$) [Klir and Yuan, 1995, 63]. The conditionals and negations derived from these three t-norms (as residuum and precomplement) are shown in Figure 15. The Łukasiewicz operations are the

$$\begin{array}{ll}
\text{Łukasiewicz:} & x \overset{\star}{\rightarrow} y = \begin{cases} 1 & \text{if } x \leq y \\ 1 - x + y & \text{if } x > y \end{cases} & \overset{\star}{\neg} x = 1 - x \\
\text{Gödel:} & x \overset{\star}{\rightarrow} y = \begin{cases} 1 & \text{if } x \leq y \\ y & \text{if } x > y \end{cases} & \overset{\star}{\neg} x = \begin{cases} 1 & \text{if } x = 0 \\ 0 & \text{otherwise} \end{cases} \\
\text{Product:} & x \overset{\star}{\rightarrow} y = \begin{cases} 1 & \text{if } x \leq y \\ y/x & \text{if } x > y \end{cases} & \overset{\star}{\neg} x = \begin{cases} 1 & \text{if } x = 0 \\ 0 & \text{otherwise} \end{cases}
\end{array}$$

Figure 15: Residua and precomplements

familiar ones—which is why the t-norm that gives rise to them is called the Łukasiewicz t-norm. Note however that this t-norm is *not* the min operation used to define conjunction in Łukasiewicz’s many-valued logics (§2.2): that min operation is the Gödel t-norm, and the conditional and negation to which it gives rise are not Łukasiewicz’s. The Gödel operations are named for their discussion in Gödel [1986]. Note that while the Gödel conjunction is the same as the conjunction in Zadeh/Kleene logic, the conditionals and negations in these logics are different. The conditional arising from the product t-norm was discussed in Goguen [1968–69] and is often referred to as the Goguen conditional.

It is common to add an additional ‘weak’ conjunction to each of these fuzzy logics—with the t-norm conjunction then being termed ‘strong’—defined

as follows (with subscripts w and s indicating strong and weak):

$$(\alpha \wedge_w \beta) =_{df} \alpha \wedge_s (\alpha \rightarrow \beta)$$

Whatever continuous t-norm is used for \wedge_s here (and where \rightarrow is its residuum), it turns out that *weak* conjunction is always the min operation [Hájek, 1998, 36]. Disjunction also comes in strong and weak forms. Weak disjunction is defined in terms of weak conjunction and conditional:

$$(\alpha \vee_w \beta) =_{df} ((\alpha \rightarrow \beta) \rightarrow \beta) \wedge_w ((\beta \rightarrow \alpha) \rightarrow \alpha)$$

It always turns out to be the max operation [ibid.]. Strong disjunction is defined as the dual of strong conjunction:

$$x \vee_s y = 1 - ((1 - x) \wedge_s (1 - y))$$

(Where the strong conjunction is a t-norm, this dual operation will always be a t-conorm. A t-conorm is a binary operation $\overset{\star}{\vee}$ on $[0, 1]$ which satisfies conditions exactly like those for a t-norm—i.e. put $\overset{\star}{\vee}$ for $\overset{\star}{\wedge}$ throughout Figure 13—*except* that we replace the final two conditions with $1 \overset{\star}{\vee} x = 1$ and $0 \overset{\star}{\vee} x = x$.) In our three systems, this definition yields the operations shown in Figure 16. The duality of strong conjunction and disjunction means that

Łukasiewicz:	$x \vee_s y = \min(1, x + y)$
Gödel:	$x \vee_s y = \max(x, y)$
Product:	$x \vee_s y = x + y - x \cdot y$

Figure 16: t-conorms/strong disjunctions

where negation is defined as $\neg x = 1 - x$ —for example, in Łukasiewicz t-norm logic—the de Morgan laws will hold (when stated in terms of negation, strong conjunction and disjunction).

Hájek [1998] introduces an axiom system for a logic BL (basic logic) which proves all and only those formulas which come out as tautologies (taking 1 as the only designated value) no matter what continuous t-norm is taken as the truth function for (strong) conjunction. Each of the three specific t-norm logics mentioned above is then axiomatized by the addition of an axiom or axioms—different in each case—to BL.

Thus far we have mentioned only propositional logics. Most many-valued semantics for propositional logic can be generalized to predicate logic by following through the following fundamental idea. We identify a subset S

of a background set U with its characteristic function f_S : a function from U to the set of truth values. Where x is an object in U and y is a truth value, $f_S(x) = y$ means that x is a member of S to degree y . So the truth values now function as values or degrees of *membership* (of objects in sets), as well as values or degrees of *truth* (of propositions). An n -place predicate is assigned an *extension* as its semantic value: a subset of the set of n -tuples of members of the domain; that is, a function from this set of n -tuples to the set of truth values. The truth value of the atomic formula Rab , for example—comprising a two-place predicate R followed by two names a and b —is then whatever value the extension of R assigns to the ordered pair comprising the referent of a followed by the referent of b . In order to give rules for assigning truth values to universally and existentially quantified formulas, we need to generalize the conjunction and disjunction operations (respectively) so that they assign values to sets—not just pairs—of values. For details, see Smith [2008, §1.2, §2.2].

2.4 Non-Truth-Functional Systems

All the systems we have looked at so far are truth-functional: the truth value of a compound proposition depends only on the truth values of its components; that is, the semantics of the connectives are given by associating them with truth functions. One notable method for assigning multiple values to propositions in a *non*-truth-functional way is that of probability logic, where values—probabilities—are assigned in accordance with the probability calculus. The value of $p \vee q$, for example, does not then depend only on the values of p and q : it also depends on the content of p and q , and in particular on whether these propositions are mutually exclusive. Interestingly, the set of tautologies of probability logic (with 1 as the only designated value) is the same as the set of tautologies of classical logic [Rescher, 1969, 186–7]. Another notable method is that of supervaluations [van Fraassen, 1966]. We begin with a three-valued model (i.e. a mapping from basic propositions to the values 0, 1 and $*$). Say that a classical model M_2 extends a three-valued model M_3 iff M_2 is exactly like M_3 except that where M_3 assigns $*$ to a basic proposition, M_2 assigns 1 or 0. The rule for assigning truth values to compound propositions is then this: $[\alpha] = 1$ (0) on M_3 iff $[\alpha] = 1$ (0) on *every* classical model which extends M_3 . So, for example, where p and q both have the value $*$, $p \wedge q$ will also have the value $*$; yet even though $\neg p$ will have the same value as q , $p \wedge \neg p$ will have the value 0. So the system is not truth-functional. A variant of supervaluationism is subvaluationism, which differs by having ‘some’ in place of ‘every’ in the rule for assigning truth values to compound propositions: $[\alpha] = 1$ (0) on M_3 iff $[\alpha] = 1$ (0)

on *some* classical model which extends M_3 . If we take 1 as the only designated value, then the (single-conclusion) consequence relation arising from the supervaluationist semantics is identical to the classical consequence relation (see e.g. Smith [2008, 82]), while that arising from the subvaluationist semantics is not (e.g. adjunction fails: $\alpha, \beta \not\vdash \alpha \wedge \beta$). However, if we move to a multiple-conclusion consequence relation, then the duality between supervaluationism and subvaluationism is restored: neither yields the classical consequence relation [Hyde, 1997]. Another variant is the degree-theoretic form of supervaluationism, which assigns values in the set $[0, 1]$. We start with a fuzzy model (i.e. a mapping from basic propositions to $[0, 1]$), introduce a measure on the set of classical extensions of this fuzzy model, and then say that the degree of truth of a compound sentence on the fuzzy model is equal to the measure of the set of classical extensions on which it is true. For further details see Smith [2008, §2.4.1].

2.5 Many-Valued Systems in the Loose Sense

Some of the motivations for many-valued systems to be discussed in §3 lead most naturally not to the idea of additional truth values, alongside the classical 0 and 1, but to the idea of propositions possessing *neither* (or *both*) of the classical values. This idea leads to what we termed at the outset many-valued systems in the loose sense. Instead of adding truth values, we stick with the classical set $\{0, 1\}$, but we alter the definition of a model. Instead of consisting in a *function* which assigns a truth value to each basic proposition, it now consists in a *relation* between the set of basic propositions and the set of truth values. Where a proposition is related to just *one* truth value we have the analogues of classical truth and falsity; where a proposition is related to *neither* truth value we have a truth value *gap*; where a proposition is related to *both* truth values we have a truth value *glut*. Allowing gaps or gluts but not both yields a three-status system; allowing both yields a four-status system. The question of how to assign truth values to compound propositions when some of their components have gaps or gluts or both may be addressed in ways analogous to any of those already considered in relation to three- or four-valued systems: no essentially new ideas are involved—simply the translation of existing ideas into the new setting (e.g. we might employ truth tables—with empty and/or multiply-filled cells in place of third and/or fourth values—or we might employ a version of supervaluationism; for further details see e.g. Smith [2008, §2.3, §2.4]). Some writers (e.g. Kripke [1975, 700, n.18]) are adamant that there is a deep *conceptual* difference between positing a third truth value and allowing truth value gaps, but in practice it is often more straightforward to formulate systems with additional values

and assignment functions rather than two values and assignment relations, and so even when the guiding idea is many-valued in the loose sense, the implementation is often many-valued in the strong sense (e.g. Blamey [1986], Langholm [1988]).

3 Uses of Many-Valued Logics

One of the earliest motivations for many-valued logics—going back to Aristotle [Łukasiewicz, 1930, 63–4]—was the consideration of modal propositions (e.g. ‘It is possible that p ’) and propositions concerning contingent future events (e.g. ‘I shall be in Warsaw at noon tomorrow’). In relation to the former, Łukasiewicz [1930, 51] saw that the semantics of the operator ‘It is possible that’ cannot plausibly be given by any of the four possible one-place two-valued truth functions; in relation to the latter, he argued that assigning ‘I shall be in Warsaw at noon tomorrow’ either 1 or 0 as truth value now conflicts with the fact that it is possible, but not necessary, that I shall be in Warsaw at noon tomorrow, and he writes that his “three-valued system of propositional logic owes its origin to this line of thought” [53]. This was also an important motivation for Peirce and MacColl (see Rescher [1969, 4–5] for references).

The consideration of probabilistic propositions, and in some cases propositions arising from quantum mechanics, was a motivation for MacColl, Zawirski, Reichenbach and Carnap (see Rescher [1969, 14–5, 184–8, 210–1] and Malinowski [1993, 66] for references).

Consideration of the paradoxes—set-theoretic (e.g. Russell’s) and/or semantic (e.g. the Liar, where it seems impossible to assign either truth value 1 or 0 to ‘This sentence is false’)—was a motivation for Bochvar, Moh Shaw-Kwei and others (see Rescher [1969, 13, 29, 207] for references).

Kleene was motivated by a consideration of partial recursive predicates. Think of such a predicate as coming with an algorithm. Given some objects as input, the algorithm terminates with the answer ‘Yes’: the predicate is true of these objects. Given some (other) objects as input, the algorithm terminates with the answer ‘No’: the predicate is false of these objects. But for some (other) objects, the algorithm does not terminate with either answer: the predicate is *undefined* for these objects. Hence Kleene’s three truth values: **t** (true), **f** (false) and **u** (undefined). A related application of many-valued logics is in the analysis of category mistakes arising from predicates which are truly applicable to some objects, falsely applicable to others, and *inapplicable* to yet other objects [Rescher, 1969, 28].

Kleene [1952, 335] also considers a different interpretation of his three values: “**t**, **f**, **u** must be susceptible of another meaning besides (i) ‘true’,

‘false’, ‘undefined’, namely (ii) ‘true’, ‘false’, ‘unknown (or value immaterial)’. Here ‘unknown’ is a category into which we can regard any proposition as falling, whose value we either do not know or choose for the moment to disregard; and it does not then exclude the other two possibilities ‘true’ and ‘false’.” This kind of *epistemic* application of many-valued logics is also found, for example, in Belnap [1977].

Many-valued logics have important applications in proof theory. For example, they can be used to show that certain axioms in a given axiomatization (e.g. of *classical* propositional logic) are *independent* of others. This application originates with Bernays; for illustrative examples see Malinowski [1993, 105–6] and Mendelson [1997, 43–5]. They can also be used for showing the *consistency* of a set of axioms (i.e. that not everything can be deduced from it). This application originates with Post [Rescher, 1969, 106].

Like classical logic, many-valued logics have applications in circuit/switching theory; for discussion and references see Malinowski [1993, 109–10] and Hähnle [2001, §8]. They have also been applied to scheduling problems [Ackermann, 1967, 78]. Many-valued logics have been employed in automated theorem-proving; for a discussion of the many further applications of many-valued theorem provers see Hähnle [1993, ch.7].

Many-valued logics have been applied to the analysis of presupposition. ‘Bill knows that Ben is late’ presupposes that Ben is late; ‘The king of France is bald’ presupposes that there is a king of France; and so on. The many-valued approach to presupposition holds that if one of a sentence’s presuppositions is false, then the sentence is neither true nor false. This is known as a *semantic* approach to presupposition, because it holds that presupposition failure affects the *truth value* of a sentence. For further details see Gamut [1991, 178–90, 212–4].

A closely related use of many-valued logics is in the treatment of singular terms which lack referents (e.g. names such as ‘Pegasus’ and definite descriptions such as ‘the king of France’). On this approach, atomic statements involving such terms are neither true nor false. This idea, which originated with Strawson [1950], was the motivation for the development of supervaluationism by van Fraassen [1966].

The final use of many-valued logics which we shall mention here is in the treatment of vagueness, where the basic thought is that statements about borderline cases of vague predicates (e.g. ‘Bob is tall’, where Bob is neither clearly tall nor clearly not tall) are neither true nor false. Supervaluationist and fuzzy approaches in particular have played an important role in the vagueness literature (see Smith [2008] for further discussion and references).

Related Topics

2.8 Intuitionism; 4.3 Presupposition and Implicature; 4.5 Compositionality; 4.9 Reference, Truth and Fiction; 4.11 Vagueness; 4.14 Empty Names.

References

- Robert Ackermann. *An Introduction to Many-Valued Logics*. Routledge & Kegan Paul, London, 1967.
- Nuel D. Belnap. A useful four-valued logic. In J. Michael Dunn and George Epstein, editors, *Modern Uses of Multiple-Valued Logic*, pages 8–37. D. Reidel, Dordrecht, 1977.
- Merrie Bergmann. *An Introduction to Many-Valued and Fuzzy Logic: Semantics, Algebras, and Derivation Systems*. Cambridge University Press, Cambridge, 2008.
- Stephen Blamey. Partial logic. In D. Gabbay and F. Guentner, editors, *Handbook of Philosophical Logic*, volume III, pages 1–70. D. Reidel, Dordrecht, 1986.
- Leonard Bolc and Piotr Borowik. *Many-Valued Logics*, volume 1. Springer-Verlag, Berlin, 1992.
- D.M. Gabbay and F. Guentner, editors. *Handbook of Philosophical Logic*, volume 2. Kluwer, Dordrecht, 2nd edition, 2001.
- L.T.F. Gamut. *Logic, Language, and Meaning*, volume I. University of Chicago Press, Chicago, 1991.
- Kurt Gödel. On the intuitionistic propositional calculus. In Solomon Feferman et al., editor, *Kurt Gödel: Collected Works*, volume I, pages 222–5. Oxford University Press, New York, 1986. Originally published in 1932.
- Joseph Goguen. The logic of inexact concepts. *Synthese*, 19:325–73, 1968–69.
- Reiner Hähnle. *Automated Deduction in Multiple-valued Logics*. Clarendon Press, Oxford, 1993.
- Reiner Hähnle. Advanced many-valued logics, 2001. In Gabbay and Guentner [2001], pages 297–395.
- Petr Hájek. *Metamathematics of Fuzzy Logic*. Kluwer Academic Publishers, Dordrecht, 1998.
- Dominic Hyde. From heaps and gaps to heaps of gluts. *Mind*, 108:641–60, 1997.
- Stanisław Jaśkowski. Investigations into the system of intuitionist logic, 1936. In McCall [1967], pages 259–63.
- Stephen Cole Kleene. *Introduction to Metamathematics*. D. Van Nostrand, Princeton, 1952.

- George J. Klir and Bo Yuan. *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. Prentice Hall, Upper Saddle River, NJ, 1995.
- Saul Kripke. Outline of a theory of truth. *Journal of Philosophy*, 72:690–716, 1975.
- Tore Langholm. *Partiality, Truth and Persistence*. CSLI, Stanford, 1988.
- Jan Łukasiewicz. Philosophical remarks on many-valued systems of propositional logic, 1930. In McCall [1967], pages 40–65.
- Jan Łukasiewicz and Alfred Tarski. Investigations into the sentential calculus, 1930. In Tarski [1956], pages 38–59.
- Grzegorz Malinowski. *Many-Valued Logics*. Clarendon Press, Oxford, 1993.
- Storrs McCall, editor. *Polish Logic: 1920–1939*. Clarendon Press, Oxford, 1967.
- Elliott Mendelson. *Introduction to Mathematical Logic*. Chapman & Hall, London, fourth edition, 1997.
- E.L. Post. Introduction to a general theory of elementary propositions. *Bulletin of the American Mathematical Society*, 26:437, 1920.
- E.L. Post. Introduction to a general theory of elementary propositions. *American Journal of Mathematics*, 43:163–85, 1921.
- Graham Priest. *An Introduction to Non-Classical Logic: From Ifs to Is*. Cambridge University Press, Cambridge, second edition, 2008.
- Nicholas Rescher. *Many-valued Logic*. McGraw-Hill, New York, 1969.
- Nicholas J.J. Smith. *Vagueness and Degrees of Truth*. Oxford University Press, 2008.
- P.F. Strawson. On referring. *Mind*, 59:320–44, 1950.
- Alfred Tarski. *Logic, Semantics, Metamathematics: Papers from 1923 to 1938*. Clarendon Press, Oxford, 1956. Translated by J.H. Woodger.
- Alasdair Urquhart. Basic many-valued logic, 2001. In Gabbay and Guenther [2001], pages 249–95.
- Bas C. van Fraassen. Singular terms, truth-value gaps and free logic. *Journal of Philosophy*, 63:481–95, 1966.

Further Reading

For readers whose only background in logic is a typical introductory course or book: Bergmann [2008] provides a thorough introduction to Bochvar, Kleene and Łukasiewicz three-valued logics, and fuzzy logics.

For readers with greater background in logic: both Malinowski [1993], and Urquhart [2001] plus Hähnle [2001], provide compact, wide-ranging surveys of the field, including important details about consequence relations, functional completeness, proof theory, decidability and complexity, and other topics of

central importance in logic; for t-norm fuzzy logics see Hájek [1998].

For readers wishing to explore applications of many-valued logics with particular relevance to philosophy of language: Smith [2008] gives an extended argument for the conclusion that the correct account of vagueness must involve a many-valued semantics.

Biographical Note

Nicholas Smith is Senior Lecturer in Philosophy at the University of Sydney. He did his undergraduate work at Sydney and his graduate work at Princeton (PhD 2001). His book *Vagueness and Degrees of Truth* was published by Oxford University Press in November 2008.

Word Count

6,055