Notre Dame Journal of Formal Logic
Volume 29, Number 1, Winter 1988

Infinite Truth-Functional Logic

THEODORE HAILPERIN

What we cannot speak about [in $\aleph_0$ or fewer propositions] we must pass over in silence.

L. Wittgenstein [emended]

1 It is oft-mentioned that universal and existential quantification are generalizations of, respectively, conjunction and (nonexclusive) disjunction. Thus, for a finite domain and in a language having a name for each individual of the domain, the universal quantification of an open sentence with one free variable is equivalent to the finite conjunction of all instances of the open sentence obtained by substituting names of the individuals for the variable. Similarly for existential quantification and disjunction. For infinite domains, or for languages which lack names for all elements of the domain, quantifiers do, implicitly, express conjunctions or disjunctions over all elements of the domain. Historically the motivation for the introduction of quantifiers was the need to express the notions of "all (individuals)" and "some (individuals)", rather than that of generalizing conjunction and disjunction. The kind of generalization of connectives achieved by quantifiers is restricted to the case of components all of which are instances of one and the same open sentence, thus depending ultimately on a subject-predicate analysis of atomic sentences and the associated notions of a domain of individuals and their properties. Conceivably the world could be viewed in other ways than exclusively in terms of individuals (objects) and their properties. Logical theory shouldn't preclude the possibility, remote as it may seem, of a language whose atomic sentences express a world view and yet have no—or no known—inner structure relating to logic. But how could there be for such a language anything beyond ordinary truth-functional (propositional, sentential) logic?

Our intention here is to explore a form of logic which, indeed, considers the analysis of sentences only into truth-functional components. It is, however, more than ordinary truth-functional logic in that it includes the notion of an infinite conjunction of sentences. Despite its rather slender conceptual basis it can nevertheless be used to do first-order quantifier logic: *Truth-functional logic,*

Received November 20, 1985; revised February 24, 1986

supplemented with the notion of infinite conjunction, suffices for the development of quantifier logic.

2 We assume we are dealing with a formal language which has a countable sequence $A_1, A_2, \ldots, A_n, \ldots$ of *atomic sentences.* The inner structure, if any, of these sentences is for the present of no concern. Using these atomic sentences as a basis, and with negation and finite and infinite conjunction, we define what a formula for such a language is. First a list of the *symbols* used:

(a) The letter 'A' with a subscript which can be of three kinds:
  (i) numerals: '1', '2', '3', ...
  (ii) indices: 'i', 'j', 'k', ...
  (iii) arithmetic function expressions (in some standard notation) having indices or numerals in argument places: e.g., '$2i - 1$', '$3i + 2j$', '$2 \cdot 5 - 1$', etc.
(b) connectives: '$\neg$', '$\wedge$'
(c) and-quantifiers: '$\bigwedge_i$', '$\bigwedge_j$', '$\bigwedge_k$', ...
(d) left and right parentheses: '(', ')'.

From now on we shall adopt that abuse of language which drops the single quotes used to form a name of what's enclosed and use symbols and expressions as names of themselves.

For *formulas* we use a recursive specification:

(a) The letter A with a subscript is a formula.
(b) If ϕ is a formula, so is $\neg\phi$.
(c) If ϕ and ψ are formulas, so is $(\phi \wedge \psi)$.
(d) If ϕ is a formula, so is $\bigwedge_i \phi$ (and similarly for any index in place of i).

We assume the other logical connectives to be defined as usual in terms of $\neg$ and $\wedge$, and that generally recognized conventions, e.g., omission of parentheses, prevail. We introduce $\bigvee_i \phi$ as an abbreviation for $\neg\bigwedge_i\neg\phi$ and refer to $\bigvee_i$ as an or-quantifier. It will sometimes be convenient to include $\vee$ and $\bigvee_i$ as official symbols of the language. The terminology *scope of a quantifier*, and *free* and *bound* for individual variables as used in ordinary quantification theory, will be taken over here and used correspondingly, with indices $i, j, k, \ldots$ playing the role of the individual variables. A *closed* formula, i.e., a *sentence*, has no free indices.

Turning to semantics, we now specify how truth values accrue to sentences from an assignment of truth values to the atomic sentences $A_1, A_2, \ldots, A_n, \ldots$. We shall use 1 (true) and 0 (false) as truth values and it will be convenient to assume that they have their numerical properties as integers. A sequence $v_1, v_2, \ldots, v_n, \ldots$ of truth values is a *model.* There are (using Cantorian counting) $2^{\aleph_0}$ models. We now specify the meaning of *a sentence σ has the value v in a model M.*

We first note that a sentence has one of the four forms:

(i) A_N, where N is either a numeral, or a number expression resulting from the substitution of numerals for indices in an arithmetic function expression.

(ii) $\neg\phi$, where ϕ is a sentence
(iii) $\phi \wedge \psi$, where ϕ and ψ are sentences
(iv) $\bigwedge_i \phi$, where ϕ is a formula having no free index except possibly i. (Here i is representative of any index.)

Conditions determining the truth value which a sentence σ has in a model M:

(i) If σ is A_N, then it has the value v_n in M, where v_n is the n'th member of the sequence M, n being the number named by the number expression N (e.g., $A_{2\cdot 5-1}$ has the value v_9).
(ii) If σ is $\neg\phi$ and ϕ has the value v in M, then $\neg\phi$ has the value $1 + v$ (modulo 2) in M.
(iii) If σ is $\phi \wedge \psi$ and ϕ and ψ have, respectively, the values v and w in M, then $\phi \wedge \psi$ has the product value vw in M.
(iv) Let σ be $\bigwedge_i \phi$ and let $\phi[n/i]$ be the sentence resulting on replacing all free occurrences of i in ϕ by the numeral n. If $\phi[n/i]$ has the value w_n in $M (n = 1,2,\ldots)$, $\bigwedge_i \phi$ has the value $\prod_{n=1}^{\infty} w_n$ in M.

A sentence σ is *true in* (*a model*) M if it has the value 1 in M; it is *valid* if true in every M. A formula is valid if the sentence which is its and-quantifier closure is valid.

It is a relatively easy matter to show that the analogues of axioms in any of the various axiomatic formulations of first-order logic are valid formulas of infinite truth-functional logic, and also that the analogues of the rules of inference preserve validity. For example, consider the axiom (schema)

(1) $\forall x\phi \rightarrow \psi$,

where ψ is the formula resulting from ϕ by replacing each free occurrence of x in ϕ by a term t which is free for x in ϕ.

Its analogue would be:

(2) $\bigwedge_i \phi \rightarrow \psi$,

where ψ is the formula resulting from ϕ by replacing each free occurrence of the index i by a subscript s which is free for i in ϕ, i.e., which is such that no index present in s would thereby lie within the scope of an and-quantifier with the same index.

To see that (2) is valid, suppose that the distinct free indices in (2) are all replaced by numerals and that the result is

(3) $\bigwedge_i \phi^* \rightarrow \psi^*$,

where ψ^* is $\phi^*[s^*/i]$ (s^*, because s may have occurrences of a free index). Now s^* names some numeral n_s, say, so that in any model the value of $\phi^*[n_s/i]$ would be the same as that of $\phi^*[s^*/i]$. Thus in any model $\bigwedge_i \phi^*$ would evaluate to a product one of whose values is the same as that of ψ^*. This implies the truth of (3) in any model. Since the substituted numerals in (3) were arbitrary we conclude that (2) is valid. As an example with an inference rule consider

(4) $\dfrac{\phi}{\forall x\phi},$

which would have as its analogue

(5) $\dfrac{\phi}{\bigwedge_i \phi}.$

This clearly preserves validity since the closures of ϕ and $\bigwedge_i \phi$ are, apart from the arrangement of the initial quantifiers, the same sentence. Thus an axiomatic formulation for deriving valid formulas of infinite truth-functional logic *using schematic letters for formulas* (as in the examples) would be identical in appearance with that of first-order quantifier logic except for the quantifier symbols.

The presence of item (iii) under (a) in the listing of symbols is significant. Without it the system would be formally the same as a trivial, one-predicate, monadic predicate calculus, with $\bigwedge_i$ in the role of $\forall x$ and A_i in the role of $A(x)$. Introduction of item (iii) under (a) provides for the capability of selecting out infinite subsequences of $A_1, A_2, \ldots, A_n, \ldots$; e.g., the derivable formula $\bigwedge_i A_i \to \bigwedge_i A_{2i}$ has no counterpart in ordinary quantifier logic.

3 Any one of a variety of completeness proofs for first-order logic may now be carried over to infinite truth-functional logic, establishing that a formula is derivable if and only if valid. However there are some matters of detail that require comment.

A form of the completeness proof well-suited to be used as a framework for our comments is the one in Chapter 31 of Quine's *Methods of Logic*, third edition. The proof begins with the prenex normal form of the negation of a closed formula (Quine's proof allows for the possibility of more than one premise, but we shall drop this generality), and derives in a prescribed manner, by repeated successive uses of universal and existential instantiation, additional formulas and, ultimately, quantifier-free instances of the matrix of the starting formula. The quantifiers are instantiated to variables (which are proxies for individuals of some as yet unspecified domain) and there can be infinitely many instances of the matrix generated by the process. The proof then goes on to show—and this is the crucial part—that either some finite subset of the matrix instances is truth-functionally inconsistent (and thereby the starting formula is false in any model), or else there is an assignment of truth values to the atomic formulas having an appearance somewhere in any of the matrix instances which makes all of the instances simultaneously true. While the proof then goes on to use this assignment to define a model (interpretation for the predicates) in which the starting formula is true, this is as far as a parallel proof for infinite truth-functional logic can go, since in place of the referred-to atomic formulas we would have occurrences of the letter A with subscripts that are arithmetic function expressions with argument places occupied by indices. Our proof would then continue by relabeling the subscripted A's as $A_1, A_2, \ldots, A_n, \ldots$ (which in no way disturbs the truth-functional relationships) and then conclude, in the case of the second alternative, that there is an assignment of values to A_1, $A_2, \ldots, A_n \ldots$ (i.e., a model) which makes all the matrix instances and then the starting formula, true. Our conclusion would be the same but with a different meaning for "formula" and for "model".

As an immediate consequence of the completeness proof for infinite truth-functional logic we have the following Löwenheim–Skolem type result.

Theorem *There is a countable set of models such that if a formula is true in each model of the set, then it is valid, i.e., true in all models.*

Proof (using the axiom of choice): By the completeness result each consistent sentence has at least one model, i.e., a model in which it evaluates to true. There are only a countable number of consistent sentences ϕ_n, $n = 1, 2, 3, \ldots$; for each select a model M_n. The set of all such M_n is the countable set of the theorem. To show this let ψ be a sentence true in all M_n. Its negation, $\neg\psi$, is different from (i.e., not logically equivalent to) any ϕ_n since there is at least one assignment of values to A_1, $A_2, \ldots, A_n, \ldots$ making one false and the other true. But $\neg\psi$, differing from every consistent sentence, must be inconsistent, and ψ then valid.

The following schematic truth-table diagram is a pictorial representation of the Theorem. (We use now T and F in place of 1 and 0.)

All consistent sentences

	A_1	A_2	$\cdot\ \cdot\ \cdot$	ϕ_1	ϕ_2	$\cdot\ \cdot\ \cdot$	$\neg\psi$	ψ
M_1	v_{11}	v_{12}	$\cdot\ \cdot\ \cdot$	T			F	T
M_2	v_{21}	v_{22}	$\cdot\ \cdot\ \cdot$		T		F	T
$\cdot$	$\cdot$	$\cdot$	$\cdot\ \cdot\ \cdot$			$\cdot$	$\cdot$	$\cdot$
$\cdot$	$\cdot$	$\cdot$	$\cdot\ \cdot\ \cdot$			$\cdot$	$\cdot$	$\cdot$
$\cdot$	$\cdot$	$\cdot$	$\cdot\ \cdot\ \cdot$			$\cdot$	$\cdot$	$\cdot$

$\aleph_0$ rows

The rows under $A_1, A_2, \ldots, A_n, \ldots$, $2^{\aleph_0}$ in number, are the models; the portion above the dashed line contains the models $M_1, M_2, \ldots$ of the theorem. The sentence $\neg\psi$, differing in value from each ϕ_n in at least one line (model) can't be a consistent sentence. Hence ψ is valid.

4 It is fairly clear that infinite truth-functional logic should support, i.e., be sufficient for carrying out logical inferences of, a first-order predicate language. One simply thinks of the atomic sentences of the first-order language as mapped onto the $A_1, A_2, \ldots, A_n, \ldots$ in such a way that those referring to a given predicate can be selected out. As an example, consider a first-order language having two singulary predicates P and Q, and a binary predicate R. To $P(x)$, $Q(x)$, and $R(x,y)$ we associate A_{3i}, A_{3i+1}, and $A_{3f(i,j)+2}$, where $f(i,j)$

is an arithmetic function establishing a one-to-one correspondence between ordered pairs of positive integers and the positive integers. The indices present in the subscripts keep track of the variables involved in the predicate expressions and the congruence classes (0, 1, or 2) of the subscripts tells which predicate is being imaged. More complicated expressions involving connectives and quantifiers are handled in obvious fashion; i.e., $\forall x(P(x) \to Q(x))$ is represented by $\bigwedge_i (A_{3i} \to A_{3i+1})$ and

$$\forall x[\exists y(P(x) \wedge R(y,x)) \to \exists y(Q(x) \wedge R(y,x))]$$

by

$$\bigwedge_i [\bigvee_j (A_{3i} \wedge A_{3f(j,i)+2}) \to \bigvee_j (A_{3i+1} \wedge A_{3f(j,i)+2})].$$

Clearly, for first-order predicate languages there is no advantage in using the symbolism of infinite truth-functional logic. But if some being comes along with a language in which there are a countable number of unanalyzed atomic sentences and wants to do logic, then we are ready for him, her or it. In a more serious vein, we see that quantifier logic can have a semantic basis (that of our Section 2) which is simpler and more ontologically neutral than that of the usual Tarskian type.

Department of Mathematics
Lehigh University
Bethlehem, Pennsylvania 18015