Notre Dame Journal of Formal Logic
Volume 25, Number 4, October 1984

Do We Need Quantification?

PHILIP HUGLY and CHARLES SAYWARD*

The standard response is illustrated by Lemmon's claim that

> ...if all objects in a given universe had names...and there were only finitely many of them, then we could always replace a universal proposition about that universe by a complex proposition. It is because these two requirements are not always met that we need universal quantification. [2], p. 105

We are partly in agreement with Lemmon and partly in disagreement. From the point of view of syntax and semantics we can replace a universal proposition about *any* universe (finite or infinite, countable or uncountable) by a complex proposition (= sentence built up from atomic sentences and the connectives). But from the point of view of communication such a replacement is not possible if the universe is infinite.

1 Skolem functions Less its quantifiers, the devices for sentence formation available in a standard first-order language are its predicates, simple names and functors, and connectives for sentential composition. Lemmon asks whether quantifications are generally dispensable in favor of sentences built up with just these latter devices. He answers, no.

The eliminability of quantification through Skolem functions appears to establish the opposite. We here sketch the situation.

As a model M of a standard first-order language L we take the structures $\langle D, v\rangle$ for nonempty domain D and valuation function v defined in standard fashion for variables, predicates, simple names and functors, if any, and for-

*We wish to thank an anonymous referee of this journal for a number of useful observations on several points on an earlier draft of this paper. In particular, we are indebted to that referee for bringing to our attention the apparent relevance of Skolem functions to Lemmon's dispensability question.

Received September 26, 1983; revised March 26, 1984

mulas built up from these sorts of symbols plus symbols for composition and quantification.

Now suppose v is defined for $\phi(\alpha,\bar{\alpha})$ in which, for $n \geq 0$, exactly the distinct variables $\bar{\alpha} = \alpha_1, \ldots, \alpha_n$ and α have free occurrences and the not necessarily distinct elements $b_1, \ldots, b_n$ are each in D. We then define a subset D^* of D relativized to $b_1, \ldots, b_n$, $n \geq 0$, and $\phi(\alpha,\bar{\alpha})$:

(1) $D^* = \{a : a \in D$ and for some v', which differs at most from v in assigning b_1 to $\alpha_1, \ldots, b_n$ to α_n and a to α, $v'(\phi) = 1\}$.

Let relation R well-order D. We now define a Skolem function for degree $n+1$ formula $\phi(\alpha,\bar{\alpha})$:

$\{\langle\langle b_1, \ldots, b_n\rangle, a\rangle$: $b_1, \ldots, b_n \in D$ and $a =$ the R-least element of D^* if that set is nonempty; otherwise $a =$ the R-least element of $D\}$.

(If $\{b_1, \ldots, b_n\}$ is null, i.e., if $n = 0$, we take a alone as the Skolem function.)

A language L^0 is a Skolem extension of a language L just in case it provides, in some form, for the expression of all of the Skolem functions for each of its degree $n+1$ formulas. The models for L^0 are, for each model $M = \langle D, v\rangle$ of L, $M^0 = \langle D, v^0, R\rangle$ for each well-ordering R of D and v^0 defined also for the symbols expressing the Skolem functions. We call L^0 a Skolem language.

The standard method of expressing Skolem functions is through *functors* of the form f_ϕ. v^0 assigns to each f_ϕ the Skolem function for its $\phi(\alpha,\bar{\alpha})$. It may be shown that the following condition is satisfied for all quantifier-free degree $n+1$ formulas of each Skolem language L^0:

(2) For each model M^0 of L^0, and each v^* which differs at most from v^0 in assigning $b_1 \in D$ to $\alpha_1, \ldots, b_n \in D$ to α_n, $v^*(\phi(f_\phi(\bar{\alpha}),\bar{\alpha})) = v^*(\Sigma\alpha\phi(\alpha,\bar{\alpha}))$.

From this it easily follows that each sentence of L^0 is equivalent to some quantifier-free sentence of L^0. So, quantification is directly dispensable in each Skolem language, and since each L has (conservative) Skolem extensions, we can say that quantification is indirectly dispensable in every non-Skolem language as well.

Does this show that Lemmon was wrong? Note that Skolem dispensability is not dispensability in favor of *just* predicates, simple names and functors, and connectives. The Skolem languages use the f symbol and *it* is not a predicate, a simple name or functor, or a connective. So, Skolem dispensability does not show that Lemmon's claim was *incorrect*.

But does such dispensability perhaps show that Lemmon's claim was *uninteresting*? After all, the basic question is whether quantifications are dispensable in favor of sentences built up from predicates, terms, and connectives. Skolem languages satisfy this dispensability criterion. They merely provide a new diversity of terms enabling dispensability.

Certainly, if a reply on behalf of Lemmon here takes the form of merely pointing out that his reduction base did not include the Skolem symbol, then Lemmon's claim is maintained by an arbitrary stipulation which does render it uninteresting.

We shall try to show that what distinguishes operations with quantifier symbols from the remaining operations in a first-order language *equally and on the same grounds* thus distinguishes operations with the Skolem symbol.

What is *central* to quantification is that it is an operation which so forms a formula ψ from a formula ϕ that

(i) there are fewer variables with free occurrences in ψ than there are in ϕ,
(ii) ϕ occurs in ψ, and
(iii) the valuation of ψ is a function of a range of valuations of ϕ.

We say that the result of replacing the free occurrences of α in $\phi(\alpha,\bar{\alpha})$ by occurrences of the Skolem term $f_\phi(\bar{\alpha})$ is an α-skolemization of ϕ. (Note that principle (2) above concerns precisely the dispensability of quantification in favor of skolemization as just defined.)

Now consider the operation of skolemization. Note that formula $\phi(\alpha,\bar{\alpha})$ has $n+1$ free variables. Its α-skolemization, $\phi(f_\phi(\bar{\alpha}),\bar{\alpha})$, has just n free variables (e.g., '$Gf_{Gxy}(y)y$' has 1 whereas 'Gxy' has 2 free variables). So, skolemization satisfies condition (i). Further, ϕ occurs in $\phi(f_\phi(\bar{\alpha}),\bar{\alpha})$ and $v(\phi(f_\phi(\bar{\alpha}),\bar{\alpha}))$ is a function of $v(f_\phi(\bar{\alpha}))$ which is a function of a range of valuations of ϕ (see (1) above). So skolemization also satisfies conditions (ii) and (iii) above.

We shall call any operation which so forms a formula ψ from a formula ϕ as to satisfy conditions (i)–(iii) above a *generalization-operator*.

To see what Lemmon was really after in considering whether *quantification* is eliminable one must see what *kind* of operation it is; for Lemmon was concerned with whether operations of that kind are eliminable. Obviously to eliminate quantification in favor of another operation of the same kind is *not* to show the eliminability of what Lemmon concluded was ineliminable. Seen in this light, and given our account of the kind of operation quantification is, we can more fully express Lemmon's question as follows:

> Is it generally possible to dispense with generalization operations in favor of the remaining devices of first-order languages?

The facts about Skolem functions sketched above plainly do not serve to counter Lemmon's negative answer to this question.[1]

2 The dispensability function On our interpretation Lemmon's question is whether it is generally possible to replace generalizations by complex sentences free of generalization devices. We now focus that question on first-order languages with just quantifiers as generalization devices, and define the general replaceability notion for those languages using the more specific terminology of *dispensability*.

We here construe a first-order language L as a countable set of names and predicates, not void of the latter. Formulas of L are built up from predicates, and perhaps names, by use of variables, connectives, and quantifier symbols. (For simplicity, we omit functors and any more than two connectives and a simple quantifier symbol 'Σ'.) An interpreted language $\underline{L}$ is a pair $\langle L, M\rangle$ where L is a first-order language and M is a model of L. By a model M of L we mean a pair $\langle D, v\rangle$ where D is a nonempty domain set and v is a valuation function defined, relative to D, for the individual symbols, predicates, and formulas of L. M is *complete* if v maps the names onto D. M is *quantificationally complete* just in case for every variable α and formula ϕ of L, $v(\Sigma\alpha\phi) = v\left(\phi\frac{\alpha}{\beta}\right)$ for some name β of L. $\left(\phi\frac{\alpha}{\beta}\right.$ is the result of replacing each free occurrence of α in ϕ by an individual constant $\left.\beta.\right)$

The dispensability of quantification in an interpreted language $\underline{L}$ will essentially consist in there being a certain effective mapping of the quantifications into the quantifier-free sentences of $\underline{L}$. Let m be some such mapping. Roughly expressed, the existence of m establishes the dispensability of quantification in $\underline{L}$ if, for each quantification ϕ of L, ϕ and $m(\phi)$ have the same truth-conditions in $\underline{L}$.

But what does having the same truth-conditions come to? Sameness of truth-conditions in $\underline{L}$ cannot be understood in terms of sameness of truth-value in $\underline{L}$, since then all sentences true in $\underline{L}$ would have the same truth-condition in $\underline{L}$. Nor can we appropriately interpret sameness of truth-conditions in $\underline{L}$ as logical equivalence. This can be seen from the following example. Let $\langle D, v\rangle$ be a complete model of L, and D be a unit set. Then quantification is clearly dispensable in $\underline{L} = \langle L, M\rangle$, and in particular by virtue of a function m such that, e.g., $m(\Sigma\alpha\phi) = \phi\frac{\alpha}{\beta}$ for each $\phi = \psi\alpha$ for unary predicate ψ of L, variable α, and some individual constant β of L. Yet no quantification is logically equivalent to an atomic sentence.

Call a function *dis* a dispensability function if and only if: (1) it is an effective mapping of the sentences of L into the quantifier-free sentences of L and (2) for any quantificationally complete model $M = \langle D, v\rangle$ of L and sentence ϕ of L, $v(\phi) = v(dis(\phi))$. If M is quantificationally complete for L, then the truth values of all sentences of L are determined by the truth values of all the atomic sentences of L. So we shall say: Quantification in $\underline{L} = \langle L, M\rangle$ is directly dispensable just in case there is a dispensability function from the sentences of L into the quantifier-free sentences of L and M is quantificationally complete for L. Quantification might be dispensable without being directly dispensable. Intuitively, quantification is dispensable over any finite domain. However, it may not be directly dispensable since the model may not be quantificationally complete. In each such case, however, there is a variant model which is quantificationally complete. Such a variant model results by adding names to L and extending the original model to these new names.

Accordingly, we say quantification is dispensable in $\langle L, M\rangle$ just in case it is directly dispensable in $\langle L, M\rangle$ or it is directly dispensable in $\langle L^*, M^*\rangle$ where L^* is a name extension of L, M^* agrees with M in assignments to elements of L, and M^* is quantificationally complete for L^*.

3 Infinite sentences Quine says this about the dispensability of quantification:

> If all the objects are named and finite in number, then quantification is of course dispensable in favor of alternation, and can be viewed as mere abbreviation. If the objects are infinite in number, on the other hand, the expansion would require an infinitely long alternation. [3], p. 91

But then Quine adds:

> [Earlier] we arrived at a view of expressions as finite sequences, in a mathematical sense; and the further step to infinite sequences is in no way audacious. It would, however, be distinctly a departure from all writings on grammar and most writings on logic, including this book, to invoke infinite expressions. [3], p. 91

If quantification is dispensable in $\underline{L} = \langle L, M \rangle$ then, for each quantification ϕ of L, $dis(\phi)$ must exist. If the domain set of the model is infinite, $dis(\phi)$ must be an infinite sentence, either an infinite alternation or infinite conjunction.

Now is it just a matter of custom that infinite expressions are not invoked? If this is all there is to it, then quantification over an infinite domain is at least sometimes dispensable. And there seems to be no more to it than that. For if expressions are sequences, then there is nothing wrong with construing some infinite sequences as expressions. Indeed, the situation almost trivially yields, e.g., infinite alternations. A standard alternation of L in Polish notation is a three-ary sequence with alternation symbol as its first coordinate and sentences of L as its second and third coordinates. The definition for an infinite alternation of L is then just this: an infinite sequence with an alternation sign as its first coordinate and for each n^{th} coordinate, $n > 1$, a sentence of L as that coordinate. Given the existence of an alternation sign and at least one sentence of L one can use set theory to prove the existence of an infinite alternation of L. Given infinitely many sentences of L one can prove the existence of uncountably many infinite alternations.

Viewing expressions as sequences we might define the wffs of a first-order language in this way:

1. $\langle \psi, \omega_1, \ldots, \omega_n \rangle$ is an atomic wff of L if ψ is an n-ary predicate letter of L and each ω_i is a variable or a name of L.
2. Every atomic wff of L is a finite wff of L.
3. $\langle \text{`}N\text{'}, \phi \rangle$ and $\langle \text{`}A\text{'}, \phi, \psi \rangle$ are finite wffs of L if ϕ and ψ are finite wffs of L.
4. $\langle \text{`}\Sigma\text{'}, \alpha, \phi \rangle$ is a finite wff of L if α is a variable and ϕ is a finite wff of L.
5. Nothing else is a finite wff of L.
6. Every finite wff of L is a wff of L.
7. $\langle \text{`}N\text{'}\phi \rangle$ and $\langle \text{`}A\text{'}\phi, \psi \rangle$ are wffs of L if ϕ and ψ are.
8. $\langle \text{`}A\text{'}, \phi_1, \ldots, \phi_n, \ldots \rangle$ is a wff of L if $n \geq 2$ and each ϕ_i is a quantifier-free wff of L.
9. Nothing else is a wff of L.

A sentence of L is, as usual, a closed wff. It is a set-theoretical consequence of this definition that L has uncountably many sentences.

4 An argument that quantification is always dispensable If it is simply a custom to restrict expressions to finite sequences, and this custom is all that stands in the way of the existence of the function *dis* in the case where the domain set is infinite, then one might well ask: what happens if this restriction is lifted? Our answer is that, for *any* standard first-order language L and model M of L, quantification is dispensable in $\langle L, M\rangle$.

Our argument is based on a result established elsewhere ([1], p. 361):

(*) For every language L and model $\langle D, v\rangle$ of L there is a set Δ of names not in L and a v^* which is an extension of v to Δ such that: (1) $\langle D, v^*\rangle$ is a model of $L^* = LU\Delta$ and for every sentence ϕ of L, $v(\phi) = v^*(\phi)$, and (2) for every quantification sentence $\Sigma\alpha\psi$ of L^*, $v^*(\Sigma\alpha\psi) = v^*\left(\psi\frac{\alpha}{\beta}\right)$ for some name β of L^*.

Let $\langle L, M\rangle$ be any interpreted language. Then there is a name extension of L^* of L and an assignment of elements of the domain set to these new names (yielding M^*) such that the truth-value of every sentence of L^* is determined by the truth-value of the atomic sentences of L^*. In other words, M^* is quantificationally complete for L^*. Both models agree in their assignments to the elements of L; so each sentence of L has the same truth value in the two models.

Suppose, now, M is quantificationally complete for L. Among the sentences of L are uncountably many infinite alternations each built up from atomic sentences of L and the connectives 'N' and 'A'. So there exists the requisite dispensability function. If M is not quantificationally complete for L there is, as we have just seen, a variant model M^* which agrees with M with respect to elements of L and which is quantificationally complete for a name extension L^* of L. Quantification is directly dispensable in either $\langle L, M\rangle$ or $\langle L^*, M^*\rangle$. So, for any interpreted language $\underline{L}$, quantification is dispensable in $\underline{L}$.

5 Why view expressions as sequences? The previous argument relies on two things: (*), which is just a formal fact, and Quine's idea that expressions are sequences in the mathematical sense, which is not any kind of fact at all, but, rather, a convention. Why adopt this convention?

The expressions of a first-order language L comprise a set $B \cup C$, where B is the set of *basic characters* of the language and C is a set of *strings* from B. For example, for some L we might have $B = \{`\Sigma', `A', `N', `a', `x', `F', `''\}$. The rest of the expressions of L, grammatical and otherwise, will comprise some set C of strings from B.

What are these basic characters and strings? What is the letter 'x', for example?

Much reasoning about numbers proceeds without in any way determining what is the number 0, the number 1, and so on. Equally, much mathematical research into order proceeds without in any way determining what is an ordered pair. In an entirely similar way much reasoning about expressions proceeds, without determining what is the letter 'x', what is the word 'horse', and so on.

At certain levels of theory, however, we need to fix answers to these questions about numbers and ordered pairs. Here there is great latitude. Numbers, ordered pairs, etc., can be variously construed. In a good sense, we can say that the notion of a number is a relative notion. For example, relative to one system 2 is $\{\{\Lambda\}\}$. Relative to another 2 is $\{\Lambda\{\Lambda\}\}$.

There is similar latitude in fixing the denotation of 'expression'. For example, where B is a finite set, membering seven basic characters, these characters could be identified with the numerals '0' through '6' or with seven planets of our solar system. Now at a certain level of theory we enunciate rules which generate infinitely many expressions. (For example, 'a negation of a sentence is a sentence'). So the set C, which consists of strings from B, is an infinite set. So, to be assured of sufficiently many strings we must regard them as abstract objects, for there is no law of physics which says of some set of concrete objects that it is infinite.

But among abstract objects we are at considerable liberty to choose which are to be strings. Quine's choice—strings are sequences—is natural enough, even if it is not the only one which could meet the requirements imposed by syntax and semantics.

Now we do not have to survey every conceivable alternative to Quine's choice in order to answer our question: Do we need quantification? This would not be possible in any case. More to the point, since the notion of an expression is a relative notion, the question ("Do we need quantification?") is really misstated. It should read: Is there a notion of an expression, suitable for syntax and semantics, relative to which quantification is always dispensable? Our answer has been that there is.

6 Tokening systems Lemmon gives the following as a reason for thinking quantification over an infinite domain is not replaceable:

> As we need the universal quantifier because we cannot write down an 'infinite conjunction', so we need the existential quantifier because we cannot write down an 'infinite disjunction'. [2], p. 111

A natural reaction to this claim is this: "Well, there are finite expressions which are also unwritable. Just consider:

> the sentence of English beginning with 'John is' followed by 1000^{1000} occurrences of 'the father of' and ending with 'someone'.

The sentence just described has never been written. Nor can it be. In the light of this it is hard to see the point of Lemmon's claim."

This natural reaction overlooks the distinction between expressions which are unwritable because of some contingent, finite upper bound on our capacity to produce and process tokens, or the availability of material thereunto, *and* the unwritability of infinite expressions for which no finite capacities and materials, possible or actual, would suffice. Writability in Lemmon's sense clearly means writability in virtue of some possible, though always finite, set of capacities and materials.

We have seen that, for any quantification $\Sigma\alpha\phi$, there exists an infinite disjunction $dis(\Sigma\alpha\phi)$. Lemmon does not deny this. He denies that $dis(\Sigma\alpha\phi)$ is writable. And, for this reason, he asserts that $\Sigma\alpha\phi$ is not replaceable by $dis(\Sigma\alpha\phi)$.

While it is true that from the point of view of syntax and semantics quantification is always dispensable, there seems something intuitively right about what Lemmon says. Does anything illuminating lie behind the intuition?

Writing an expression is one way of tokening it. Rules for writing expressions comprise a tokening system for a language. In practice, these rules are confused with syntactical rules. For example, instead of

$$\langle `N', \phi\rangle \text{ is a wff if } \phi \text{ is a wff,}$$

we find

$$\ulcorner N\phi \urcorner \text{ is a wff if } \phi \text{ is a wff}$$

which says

the result of writing '*N*' and then ϕ is a wff if ϕ is a wff.

Now the result of writing a wff is a perceptible particular—a concrete object. A wff is a string of basic characters. As such it must be construed as an abstract object since there are infinitely many strings.

A tokening system from a set B of basic characters is a pair $\langle B \cup C_i, t_i\rangle$ where C_i is some set of strings from B and t_i is a function with domain $B \cup C_i$ and whose range is a set of tokening instructions, i.e., instructions for constructing perceptible particulars. We regularly write $\langle B_i, t_i\rangle$ for $\langle B \cup C_i, t_i\rangle$.

As an example let $B = \{`\Sigma', `N', `A', `'', `F', `a', `x'\}$ and let C_1 be the set of all finite strings from B. Then a tokening system $\langle B_1, t_1\rangle$ for $B \cup C_1 = B_1$ can be determined by a finite set of general rules from which a specific tokening instruction for each $x \in B_1$ can be derived, as follows. Let R1–R7 be rules telling us how to token each element of B. Then we have

R8 To token $\langle s_1, \ldots, s_n\rangle \in C_1$ first token $\langle s_1, \ldots, s_{n-1}\rangle$ and then follow it a small but discernible distance to the right by the result of tokening s_n.

Being a token of is a four-place relation relating perceptible particular, expression, language, and tokening system. Accordingly, Lemmon's claim that infinite alternations are unwritable comes to this: If an infinite alternation belongs to a set of strings, no tokening system exists for that set.

Now what has this claim got to do with the dispensability of quantification? Well, consider this restraint on what is to count as an expression of first-order language L.

(α) No string of basic characters of L is an expression of L unless it is tokenable.

We have already seen that semantic and syntactic theory best construe strings as abstract objects. But, from the point of view of communication, expressions are not merely that. If an expression is something which a speaker can communicate to a hearer, then no abstract object is an expression except as it is an element of a system connected with some method by which the elements of that system can be, e.g., written down.

So Lemmon's point can perhaps be put this way: If the domain is infinite, then, while $dis(\Sigma\alpha\phi)$ might be constructible as a set-theoretic object, it is not tokenable. Consequently, it could not replace $\Sigma\alpha\phi$ for the purposes of communication. Briefly, quantification is not always replaceable by an *expression* built up solely from atomic sentences and connectives.

Is this right, even accepting (α)? Let C_2 be C_1 plus all sentences $dis(\phi)$, where ϕ is a finite sentence belonging to C_1. Among the sentences $dis(\phi)$ are infinite strings. (If ϕ is quantifier-free just set $dis(\phi) = \phi$.) Now pretty clearly there is a tokening system for $B \cup C_2 = B_2$:

$$\langle B_2, t_2 \rangle.$$

Just let t_2 be determined by R1–R8 plus this additional rule:

R9 For each finite sentence $\phi \in C_1$ where $\phi \neq dis(\phi)$, token $dis(\phi)$ by overbarring the result of tokening ϕ.

Thus, this series of marks

$$\Sigma xFx$$

tokens 'ΣxFx' $(= \langle `\Sigma', `x', `F', `x' \rangle)$ and this series of marks:

$$\overline{\Sigma xFx}$$

tokens dis('ΣxFx'), an infinite alternation.

This makes it appear that Lemmon is wrong. Infinite alternations are writable.

7 ***A possible reply*** Lemmon might have made the following reply to this argument:

"Your argument is based on your notion of a tokening system, which is defective. For consider that all you require is a finite set of rules which associate with each element of a set of expressions a unique instruction for constructing a unique concrete particular. Well, then, let X be any countable set of expressions and e any enumeration of its elements. Then consider:

(X) For each $x \in X$ token x by constructing a physical line segment $e(x)$ cm long.

This rule yields a tokening system by your definition. But do we want to say a physical line segment tokens an expression? Surely not. A hearer cannot tell what element of X is tokened just by looking at a line, say, 8 cm long.

"A tokening instruction does not merely correlate an expression with a possible concrete object. The object, in order to be a token, must render the structure of the expression perceptible.

"Returning to your overbar example, the series of marks

(**) $\overline{\Sigma xFx}$

no more tokens an infinite disjunction than does a physical line segment token an element of X. In neither case does the particular constructed perceptualize the structure of the expression it is supposed to token."

We are not persuaded. For suppose a speaker and hearer share certain knowledge. They know which expressions are in X; they know the function e; and they know rule (X). Then communication between them can take place by producing and processing line segments. The case is similar with our overbar example. Given a knowledge of the elements of B_2, a knowledge of R1–R9, a knowledge of *dis*, communication can take place by means of such items as (**) above.

It is true that communication would not proceed very easily using overbars or line segments. But it could proceed. That is all that is relevant here.

8 A further constraint on infinite expressions Earlier we made the point that just as we are free to variously construe what gets quantified over when we quantify over numbers so we are free to variously construe what gets communicated when we communicate expressions. We normally take '153' as a finite string of three basic characters, '1', '5', and '3'. But it could also be taken as an infinite string, e.g.,

$$\langle \ldots, \text{'0'}, \text{'0'}, \text{'1'}, \text{'5'}, \text{'3'} \rangle.$$

Compare these two tokening systems:

> *System* 1. Here the set of basic characters is $A = \{\text{'0'}, \ldots, \text{'9'}\}$. The set of strings, C_3, is the set of all sequences $\langle s_1, \ldots, s_n \rangle$ where $s_1 \neq \text{'0'}$ and $s_i (0 < i < n)$ is a basic character. The function t_3 is determined by this set of rules: First, there are ten rules saying how to token each basic character. Then there is the usual recursive rule:
>
> > To token $\langle s_1, \ldots, s_n \rangle \in C_3$ first token $\langle s_1, \ldots, s_{n-1} \rangle$ and follow it a small but discernible distance to the right by the result of tokening s_n.
>
> *System* 2. The set of basic characters is the same. The set of strings, C_4, is the set of all sequences $\langle \ldots, \text{'0'}, \text{'0'}, s_1, \ldots, s_n \rangle$ with the same constraints on s_1 and each s_i. The tokening rules are the same except that 'To token $\langle \ldots \text{'0'}, \text{'0'}, s_1, \ldots, s_n \rangle \in C_4$' replaces 'To token $\langle s_1, \ldots, s_n \rangle \in C_3$' in the statement of the recursive rule.

Relative to system 1 the series of marks

153

tokens a finite string; relative to system 2 it tokens an infinite string.

While the two systems $\langle A_3, t_3 \rangle$ and $\langle A_4, t_4 \rangle$ are distinct, the distinction makes no difference. Isomorpohism is a technical concept that captures this idea. Two tokening systems $\langle B_i, t_i \rangle$ and $\langle B_j, t_j \rangle$ are isomorphic if and only if there is a 1–1 function f from the union of B_i and the range of t_i onto the union of B_j and the range of t_j such that

$$t_i(x) = y \text{ iff } t_j(f(x)) = f(y).$$

Now note that there is such a function from $A_3 \cup$ the range of t_3 onto $A_4 \cup$ the range of t_4. So the two tokening systems are isomorphic.

Here, then, is the situation. The same tokening capacity by which speakers and hearers process such marks as

153

can be formally represented in infinitely many ways. Relative to some of these representations an infinite string is tokened; relative to others a finite string is tokened. All of these representations are isomorphic to one in which each string from {'0'..., '9'} is finite. This fits with the solid intuition that '153' is a finite expression.

These considerations motivate the following constraint on what is to count as an infinite expression of a first-order language L. Let B be the set of basic characters of L and C_i a set of strings from B. Then:

(β) A string $s \in C_i$ is a finite expression of L iff s is a finite string, or else, for any tokening system $\langle B_i, t_i \rangle$ there is an isomorphic tokening system $\langle B_j, t_j \rangle$ in which each $s \in C_j$ is a finite string.

An infinite string $s \in C_i$ is an infinite expression just in case there is a tokening system $\langle B_i, t_i \rangle$ which is isomorphic to no tokening system $\langle B_j, t_j \rangle$ where each string $s \in C_j$ is finite.

Returning now to our overbar example, recall these sets:

B = the set of basic characters of L
C_1 = the set of all finite strings from B
$C_2 = C_1 \cup \{x : x = dis(\phi)$ where $\phi \neq dis(\phi)$ and ϕ is a finite sentence of $L\}$
t_1 = the function determined by R1–R8
t_2 = the function determined by R1–R9.

Now dis('ΣxFx') is an infinite disjunction belonging to C_2. It is also tokenable since $\langle B_2, t_2 \rangle$ is a tokening system. But from these two facts it does not follow that dis('ΣxFx') is an infinite expression of L. For the tokening system $\langle B_2, t_2 \rangle$ is isomorphic to the tokening system $\langle B_1, t_1 \rangle$ and each string belonging to C_1 is finite.

9 Is there a tokening system for an uncountable set? Reflecting on what Lemmon's claim has to do with the general issue of whether quantification is always dispensable, we have been forced to get a clearer idea of when an expression is infinite. We have seen that whether a string s belonging to a set C_i of strings of basic characters B of L is an infinite expression of L depends not so much upon s itself as it does upon $B_i = B \cup C_i$. In particular, B_i has to be uncountable. This is true because of the following set-theoretic truth:

> If B_i is a countable set of basic characters of L and strings of basic characters of L, and $\langle B_i, t_i \rangle$ is a tokening system, then there exists a set C_j of finite strings from B and a tokening system $\langle B_j, t_j \rangle$ which is isomorphic to $\langle B_i, t_i \rangle$.

Consequently, if an infinite string is to be an infinite expression of L it has to belong to an uncountable set C of strings. And, of course, there has to be a tokening system for that set C.

Now we argue:

There is no tokening system for an uncountable set.

We do not deny the existence of a pair $\langle B \cup C, t\rangle$ where $B \cup C$ is uncountable. We deny that such a pair is a tokening system. While every tokening system is a pair of a certain sort, the reverse is not always true.

All reasoning that is to be communicated requires the repetition of symbols. This in turn requires the multiple tokening of the same symbol. For example, communication of a modus ponens inference requires each of two sentences to be twice tokened. So for $\langle B \cup C, t\rangle$ to be a tokening system each instruction $t(x)$ must be such that it is possible to determine if two concrete particulars token the same element $x \in B \cup C$.

The essentials of the situation can be most easily seen by considering the case in which each specific instruction $t(x)$ is derived from a single general rule. Such a rule correlates a variable property which discriminates the elements of $B \cup C$ with a variable perceptual property. If $B \cup C$ is denumerable and is enumerated by e, this rule would be an example:

To token an element $x \in B \cup C$ draw a line segment $e(x)$ cm long.

The variable property differentiating the elements of $B \cup C$ is the property of being the i^{th} element of $B \cup C$ relative to e. The variable perceptual property is being a (physical) line segment i cm long.

Suppose now $B \cup C$ is uncountable. Then the variable property discriminating the elements of $B \cup C$ will be continuous. The variable perceptual property will be a continuum property. This means that though we could effectively decide in certain cases that a pair of objects token different elements of $B \cup C$, we could in no case determine that a pair of objects token the same element of $B \cup C$. And this would not be because of lack of sensitivity in our measuring instruments. No increase in the sensitivity of our measuring instruments would resolve the problem.

An example will help. Suppose C is the set of all infinite decimals from B, which consists of the ten digets plus the decimal point. Then let t be determined by this rule:

To token $\langle s_1, \ldots\rangle$ construct a line segment r cm long where r is the real number represented by $\langle s_1, \ldots\rangle$

Now suppose the difference between lengths n and m is the least difference we can detect (by whatever instruments we have). There will be uncountably many lengths between n and m. Thus, given that our best instruments cannot detect a difference in length between some pair of line segments, it remains possible that they are different in length (only undetectably so). And, since there are uncountably many lengths between any two lengths, no increase in the sensitivity of our measuring instruments can resolve the problem. So while we could decide in certain cases that a pair of line segments are of *different* lengths and hence token *different* infinite decimals, we could in no case determine that a pair of line segments are of the same length and hence token the same infinite decimal. For this reason there is no tokening system for the set of infinite decimals.

It would seem that any tokening system for uncountably many syntactic items would need to exploit some continuously variable property or set of such properties so that the "indistinguishability problem" would arise for each such system. It thus appears that there is not only no tokening system for the set of infinite decimals, there is also no tokening system for any uncountable set of syntactic items.

10 Concluding remarks Consider any interpreted language $\underline{L} = \langle L, M \rangle$. Let B be the set of basic characters of L and C be the set of all strings of these basic characters. Suppose M is quantificationally complete for L. Then there exists a 1-1 function *dis* from the L-sentences of C to the quantifier-free L-sentences of C such that ϕ is true in M if and only if $dis(\phi)$ is true in M. In this case quantification is directly dispensable in $\underline{L}$.

Suppose M is not quantificationally complete for M. Then there exists an interpreted language $\underline{L}^* = \langle L^*, M^* \rangle$ such that

(i) L^* is a name extension of L
(ii) M^* agrees with M with respect to L
(iii) M^* is quantificationally complete for L^*.

Also there exists a 1-1 function *dis* from the sentences of L^* onto the quantifier-free sentences of L^* such that, for all sentences ϕ of L^*, ϕ is true in M^* if and only if $dis(\phi)$ is true in M^*. In this case quantification is directly dispensable in L^*.

So, from the point of view of syntax and semantics, quantification is dispensable in $\underline{L}$, for any interpreted language $\underline{L}$. This observation is worth noting because it is frequently supposed, in philosophical discussions at any rate, that *syntactic* and *semantic* factors are what secure the need for quantification in connection with infinite domains.

Despite this, it turns out that Lemmon is right about dispensability, though for a reason he only hints at in his reference to *writability*. Following up on this reference we have seen that the bar to dispensing with quantification in connection with infinite domains pertains not to syntactical or semantical conditions, but to conditions of *communication*, for which we have supplied the outlines of a formal representation in our notion of a tokening system.

NOTE

1. To make even plainer the clearly evident kindredness of the use of the Skolem symbol to that of quantification symbols, note that an equivalent way of exploiting Skolem functions is to give f the role of a *variable-binding* term forming operator. The formation rule is this: for each degree $n \geq 1$ formula ϕ in which variable α is free, $f\alpha\phi$ is a degree $n-1$ term. Each $f\alpha\phi$ is called a Skolem term for ϕ relative to α. Note that α is bound in $f\alpha\phi$ and bound by $f\alpha$. There is an *exact* syntactic parallel to the usual use of the iota symbol 'ι'. The differences are valuational. $f\alpha\phi$ is so evaluated as to *express* the relevant Skolem function (but not, as in the case of the Skolem functors, to *denote* them). The definition is as follows:

$v(f\alpha\phi)$ = the R-least element of $\{a : a \in D$ and, for some v' which differs from v at most in assigning a to $\alpha, v'(\phi) = 1\}$ if that set is nonempty; otherwise, $v(f\alpha\phi)$ is the R-least element of D.

REFERENCES

[1] Hugly, P., and C. Sayward, "Indenumerability and substitutional quantification," *Notre Dame Journal of Formal Logic*, vol. 23, no. 4 (1982), pp. 358–366.

[2] Lemmon, E. J., *Beginning Logic*, Hackett, Indianapolis, Indiana, 1978.

[3] Quine, W. V., *Philosophy of Logic*, Prentice-Hall, Englewood Cliffs, New Jersey, 1970.

Department of Philosophy
University of Nebraska
Lincoln, Nebraska 68588-0321