Notre Dame Journal of Formal Logic
Volume 31, Number 3, Summer 1990

Expressive Completeness and Decidability

GEORGE F. SCHUMM and STEWART SHAPIRO*

Abstract Under what conditions is the expressive completeness of a set of connectives decidable? The answer is shown to depend crucially upon how the set is encoded as input to a Turing machine.

After Żliński showed in 1924 that | and ↓ are the only dyadic connectives adequate for expressive completeness, one would have thought that there was nothing left to say on the topic. Well, almost. Let **C** be a set of (two-valued) truth-functional connectives. Under what conditions on **C** is it decidable whether **C** is expressively complete? And under what conditions is it decidable whether a given connective is definable in terms of the members of **C**?

If only because of cardinality considerations, there is no decision procedure for expressive completeness for all **C**. In the special case where **C** is finite, however, there is such a procedure. This follows from an old, and relatively obscure, result of Post's. Given the idiosyncratic nature of the work in which the result is buried, it might be useful to restate it here.

Two rows i and j of a truth table are *mirror images* if every variable that is T on i is F on j, and vice versa. The table is then *self-dual* if any two lines that are mirror images have different outputs.

A variable p is *redundant* in a given table if any two rows that differ only in the value assigned to p have the same output. Likewise, p is *decisive* if any two rows that differ only in the value assigned to p have different outputs. The table is then *alternating* if every variable is either redundant or decisive.

Finally, a truth table is *pairwise local* if, for any row i with output T and row j with output F, there is a variable that is T on i and F on j.

Theorem 1 (Post [2]) **C** *is expressively complete if and only if, for each of the following conditions, there is a connective in* **C** *whose truth table satisfies that condition*:

*Thanks are due an anonymous referee for bringing the work of Kudielka and Oliva [1] and Post [2] to our attention. The strategy of our second proof of decidability is essentially sketched in [1].

Received September 28, 1988; revised September 7, 1989

(1) *The output of the first row is* F
(2) *The output of the last row is* T
(3) *The table is not self-dual*
(4) *The table is not alternating*
(5) *The table is not pairwise local.*

There is, however, another more insightful way of getting this decidability result, which answers our second question as well. Let $L_{\mathbf{C}}$ be the language built up recursively from the set **C** of connectives. Define the *degree* of a formula of $L_{\mathbf{C}}$ as follows: if p is a variable, $\mathrm{dg}(p) = 0$; for $c \in \mathbf{C}$, $\mathrm{dg}(c(A_1, \ldots, A_n) = \max\{\mathrm{dg}(A_1), \ldots, \mathrm{dg}(A_n)\} + 1$. In the obvious fashion, we can think of formulas in n distinct variables as expressing truth functions of n arguments. We let Ω_i be the set of truth functions having less than or equal to n arguments expressed by formulas of $L_{\mathbf{C}}$ of degree less than or equal to i.

Lemma *If* $\Omega_i = \Omega_{i+1}$, *then* $\Omega_i = \Omega_j$ *for all* $j \geqq i$.

Proof: Suppose that $\Omega_i = \Omega_j$ and induct on j. This is trivial if $j = i$. Now suppose that the truth function f having less than or equal to n arguments is expressed by a formula A of degree less than or equal to $j + 1$. Then $A = c(A_1, \ldots, A_m)$ for some $c \in \mathbf{C}$ and $A_1, \ldots, A_m$ of degrees less than or equal to j and expressing truth functions having less than or equal to n arguments. By the inductive hypothesis, there are formulas $B_1, \ldots, B_m$ of degrees less than or equal to i that express the same truth functions as $A_1, \ldots, A_m$, respectively. Let C_g be the result of substituting, for each k, the kth variable of A_g for the kth variable of B_g. Then $c(C_1, \ldots, C_m)$ is of degree less than or equal to $i + 1$ and expresses f. Hence, by our initial hypothesis, some formula of degree less than or equal to i also expresses f. So $\Omega_i = \Omega_{j+1}$.

Theorem 2 *An n-argument truth function f is expressed by a formula of* $L_{\mathbf{C}}$ *if and only if f is expressed by a formula of* $L_{\mathbf{C}}$ *of degree less than or equal to*

$$d = \sum_{i=1}^{n} 2^{2^i}.$$

Proof: Suppose that f is expressed by no formula of degree less than or equal to d. Then, since there are only $d + 1$ truth functions of less than or equal to n arguments and Ω_0 is nonempty, $\Omega_i = \Omega_{i+1}$ for some $i \leqq d$. By the Lemma, $\Omega_i = \Omega_j$ for all $j \geqq i$. But then no formula expresses f.

This yields as a

Corollary **C** *is expressively complete if and only if some two-variable formula of* $L_{\mathbf{C}}$ *of degree less than or equal to twenty expresses the Sheffer stroke function.*

In the finite case, there are but finitely many two-variable formulas of $L_{\mathbf{C}}$ (modulo alphabetic relettering) of degree less than or equal to twenty. Thus, to decide the expressive completeness of **C**, all (!) one has to do is check these formulas to determine whether one of them expresses the Sheffer stroke function. Likewise, if **C** is finite, one can decide whether a given n-place connective is definable in terms of the members of **C** by checking all formulas of $L_{\mathbf{C}}$ of degree less than or equal to d. We also have an answer to an obvious query raised by Post's work in [2]. Post was not concerned with questions of decidability but rather with classifying the closed sets of expressively complete connectives. Theorem 2 shows that in the finite case it is decidable to which of his categories a given set of connectives belongs.

There is a delicate issue lurking here deserving comment. Strictly speaking, the standard notion of decidability applies only to sets of strings of '|'s and blanks. Such strings, after all, are the items that serve as inputs and outputs of Turing machines (see [3] and [4]). Now the foregoing decidability results have been cast instead in terms of sets of connectives. This is because it is straightforward to associate a finite set of truth tables with a string of the requisite sort. Write the rows of all the tables in a single line, separating rows with commas and tables with semi-colons, and then let '|', '||', '|||', and '||||' encode T, F, ', ', and ';', respectively. To be explicit, however, one should say that (e.g.) Post showed that the set of strings that correspond to finite, expressively complete sets of connectives is decidable.

The important feature of this way of converting a finite set of connectives into a string is that one has an effective procedure for recovering the semantics of each connective *and* it is possible to determine when one is finished with the set. This in turn means that there is an effective procedure for telling when one is through checking which truth functions are expressed by formulas up to a certain degree. It is clear that one cannot specify an *infinite* set of connectives in the same way, and this at bottom is why these decidability results hold only for finite sets.

In discussions of decidability, it is so readily taken for granted that finite sets are given via some such canonical specification that it is worth considering what would happen if a looser method of converting a set of connectives into a string were used. The natural candidate is a recursive specification, where one gives a decision procedure for determining whether any given connective is in the set. For example, we might let our specification consist of a string of '|'s representing in the usual fashion the index of a Turing machine. With such a specification, the foregoing proofs all fail. (For instance, under these conditions there is, in general, no way of determining that all formulas of degree less than or equal to twenty have been checked.) In fact, decidability is lost altogether:

Theorem 3 *There is no effective procedure that, given a recursive specification of a finite set of connectives, decides whether the set is expressively complete. A fortiori, the set of recursive specifications of expressively complete sets of connectives is not recursive, nor is the set of recursive specifications of finite, expressively complete sets of connectives.*

Proof: Suppose for a *reductio* that M were an effective procedure that, given a recursive specification of a finite set of connectives, decides whether the set is expressively complete. For $n \geqq 2$, let $|^n$ be the n-place Sheffer stroke. That is, the truth table for $|^n$ has T as output on its last row and F on all other rows. Where M_i is the ith entry in an effective enumeration of all Turing machines, put

$$\mathbf{C}_i = \{|^n\colon M_i \text{ halts at the } n\text{th step for input } i\}.$$

$\mathbf{C}_i$ is finite (indeed, it is either empty or the singleton $\{|^n\}$ for some n) and therefore has a recursive specification σ_i. Moreover, since $\mathbf{C}_i$ is decidable by a Turing machine, such a σ_i can be found in an effective manner. But now a set $\mathbf{C}$ of connectives is expressively complete if $|^n \in \mathbf{C}$, since $|^n(A, \ldots, A, B)$ expresses the same truth function as $|(A, B)$. It follows that $\mathbf{C}_i$ is expressively complete if and only if M_i eventually halts for input i. The assumed procedure could thus be used to solve the self-halting problem. To determine whether M_i eventually halts for input i, simply apply M to σ_i.

REFERENCES

[1] Kudielka, V. and P. Oliva, "Complete sets of functions of two and three binary variables," *IEEE Transactions on Electronic Computers*, vol. EC-15 (1966), pp. 930-931.

[2] Post, E. L., *The Two-valued Iterative Systems of Mathematical Logic*, Princeton University Press, Princeton, New Jersey, 1941.

[3] Shapiro, S., "On the notion of effectiveness," *History and Philosophy of Logic*, vol. 1 (1980), pp. 209-230.

[4] Shapiro, S., "Acceptable notation," *Notre Dame Journal of Formal Logic*, vol. 23 (1982), pp. 14-20.

Department of Philosophy
Ohio State University
Columbus, Ohio 43210