THE DECIDABILITY OF LANGUAGES THAT ASSERT MUSIC *

MICHAEL KASSLER

The construction of formalized languages that assert music calls forth investigations of these languages' decidability. Having for such a language an effective procedure that determines given any musical composition whether it is or is not an assertion of the language is obviously desirable, for (amongst other reasons) the decision procedure produces automatically a derivational analysis, in accordance with the language, of any given assertable composition. In this note we show that any formalized language of the sort we have found useful for asserting music is decidable.[1]

Such a language has the following two properties which are additional to those required of formalized languages in general. Because

* This research has been supported in part by a grant from the John S. Guggenheim Memorial Foundation.

[1] These languages include: the systems R (under principal interpretation), treated in my "Toward a Theory that is the Twelve-Note-Class System," PNM, Vol. 5, No. 2 (1967) pp. 1–80, where also the concept of derivational analysis is described and illustrated; the systems S_1 and S_2 explicating (under principal interpretation) the middleground of Heinrich Schenker's theory of tonality (for S_1 see the relevant part of my *A Trinity of Essays,* Ph.D. dissertation, Princeton University, 1967, University Microfilms order number 68–2490; for an indication of S_2, though not S_2 itself, which has not yet been published, see *Proving Musical Theorems I: The Middleground of Heinrich Schenker's Theory of Tonality,* Technical Report No. 103, Basser Department of Computer Science, The University of Sydney, 1975, and more generally, "Explication of the Middleground of Schenker's Theory of Tonality," to be published in 1977 in *Miscellanea Musicologica: Adelaide Studies in Musicology);* and the as yet unpublished sequence of systems K (under principal interpretation) intended to explicate the theory of tonality proposed by A. F. C. Kollmann.

primitive symbols or strings of them represent notes or note-classes or other basic constituents of musical compositions, and there are only finitely many such constituents,

The number of primitive symbols is finite. (1)

As such a language's rules of inference refer only to content relationships of premisses and conclusion, and never to the ordinal numbers of well-formed formulas (wffs) in a proof nor to relations (such as 'is the immediately preceding wff in the proof') having such numbers in their domains or converse domains, non-first occurrences of wffs in proofs are superfluous. That is, every theorem has a proof in which no wff appears more than once. Moreover, since a derivational analysis of a composition essentially reduces the comparatively complex 'surface structure' of the composition itself to the comparatively simple structures asserted by the axioms of the language, the intermediate stages in the analysis can be expected usually to have less structure than the composition itself. In any case, the amount of structure is the intermediate stages can safely be bounded, so that

> A computable function f is known such that if **T** is a theorem then **T** has a proof in which each constituent wff occurs only once and has at most $f(t)$ occurrences of primitive symbols, where t is the number of (occurrences of) primitive symbols in **T**. (2)

For the formalized languages we have used to assert music, setting—for all t—$f(t)$ to the larger of t^2 and 1000, would suffice.

A decision procedure for such a language L is as follows. A wff **W** of L is given. Let A be the set of all formulas of L having at most $f(w)$ occurrences of primitive symbols, where w is the number of (occurrences of) primitive symbols in **W**. Property (1) ensures that A is a finite set. Form A' from A by applying the formation rules to each element of A and discarding all formulas that are not well-formed. Let Q be the set of all sequences of wffs in A' such that in each sequence in Q **W** is the last wff and and no wff occurs more than once. This last restriction ensures that Q is a finite set. Now examine each sequence in Q in turn. If any such sequence is a proof in L— since L is a formalized language there is an effective procedure to determine this—then **W** is a theorem. If all such sequences have been examined and none is a proof then, according to property (2), **W** is not a theorem.

Of course such a decision procedure, which makes no use of the structure of the axioms or the rules of inference of L, will be inefficient, so for particular practical applications it will be useful to construct a specific less general decision procedure. This note has established that the construction of some decision procedure will always be possible.