Nonconcatenative Finite-State Morphology

by

Martin Kay

Xerox Palo Alto Research Center
3333 Coyote Hill Road
Palo Alto, CA 94304, USA

In the last few years, so called *finite-state* morphology, in general, and *two-level* morphology in particular, have become widely accepted as paradigms for the computational treatment of morphology. Finite-state morphology appeals to the notion of a finite-state transducer, which is simply a classical finite-state automaton whose transitions are labeled with pairs, rather than with single symbols. The automaton operates on a pair of tapes and advances over a given transition if the current symbols on the tapes match the pair on the transition. One member of the pair of symbols on a transition can be the designated null symbol, which we will write ε. When this appears, the corresponding tape is not examined, and it does not advance as the machine moves to the next state.

Finite-state morphology originally arose out of a desire to provide ways of analyzing surface forms using grammars expressed in terms of systems of ordered rewriting rules. Kaplan and Kay (in preparation) observed, that finite-state transducers could be used to mimic a large class of rewriting rules, possibly including all those required for phonology. The importance of this came from two considerations. First, transducers are indifferent as to the direction in which they are applied. In other words, they can be used with equal facility to translate between tapes, in either direction, to accept or reject pairs of tapes, or to generate pairs of tapes. Second, a pair of transducers with one tape in common is equivalent to a single transducer operating on the remaining pair of tapes. A simple algorithm exists for constructing the transition diagram for this composite machine given those of the original pair. By repeated application of this algorithm, it is therefore possible to reduce a cascade of transducers, each linked to the next by a common tape, to a single transducer which accepts exactly the same pair of tapes as was accepted by the original cascade as a whole. From these two facts together, it follows that an arbitrary ordered set of rewriting rules can be modeled by a finite-state transducer which can be automatically constructed from them and which serves as well for analyzing surface forms as for generating them from underlying lexical strings.

A transducer obtained from an ordered set of rules in the way just outlined is a *two level* device in the sense that it mediates directly between lexical and surface forms without ever constructing the intermediate forms that would arise in the course of applying the original rules one by one. The term *two-level morphology*, however, is used in a more restricted way, to apply to a system in which no intermediate forms are posited, even in the original grammatical formalism. The writer of a grammar using a two-level formalism never needs to think in terms of any representations other than the lexical and the surface ones. What he does is to specify, using one formalism or another, a set of transducers, each of which mediates directly between these forms and each of which restricts the allowable pairs of strings in some way. The pairs that the system as a whole accepts are those are those that are rejected by none of the component transducers, modulo certain assumptions about the precise way in which they interact, whose details need not concern us. Once again, there is a formal procedure that can be used to combine the set of transducers that make up such a system

into a single automaton with the same overall behavior, so that the final result is indistinguishable form that obtained from a set of ordered rules. However it is an advantage of parallel machines that they can be used with very little loss of efficiency without combining them in this way.

While it is not the purpose of this paper to explore the formal properties of finite-state transducers, a brief excursion may be in order at this point to forestall a possible objection to the claim that a parallel configuration of transducers can be combined into a single one. On the face of it, this cannot generally be so because there is generally no finite-state transducer that will accept the intersection of the sets of tape pairs accepted by an arbitrary set of transducers. It is, for example, easy to design a transducer that will map a string of x's onto the same number of y's followed by an arbitrary number of z's. It is equally easy to design one that maps a string of x's onto the same number of z's preceded by an arbitrary number of x's. The intersection of these two sets contains just those pairs with some number of x's on one tape, and that same number of y's followed by the same number of z's on the other tape. The set of second tapes therefore contains a context-free language which it is clearly not within the power of any finite-state device to generate.

Koskenniemi overcame this objection in his original work by adopting the view that all the transducers in the parallel configuration should share the same pair or read-write heads. The effect of this is to insist that they not only accept the same pairs of tapes, but that they agree on the particular sequence of symbol pairs that must be rehearsed in the course of accepting each of them. Kaplan has been able to put a more formal construction on this in the following way Let the empty symbols appearing in the pairs labeling any transition in the transducers be replaced by some ordinary symbol not otherwise part of the alphabet. The new set of transducers derived in this way clearly do not accept the same pairs of tapes as the original ones did, but there is an algorithm for constructing a single finite-state transducer that will accept the intersection of the pairs they all accept. Suppose, now, that this configuration of parallel transducers is put in series with two other standard transducers, one which carries the real empty symbol onto its surrogate, and everything else onto itself, and another transducer that carries the surrogate onto the real empty symbol, then the resulting configuration accepts just the desired set of languages, all of which are also acceptable by single transducers that can be algorithmically derived form the originals.

It may well appear that the systems we have been considering properly belong to finite-state phonology or graphology, and not to morphology, properly construed. Computational linguists have indeed often been guilty of some carelessness in their use of this terminology. But it is not hard to see how it could have arisen. The first step in any process that treats natural text is to recognize the words it contains, and this generally involves analyzing each of them in terms of a constituent set of formatives of some kind. Most important among the difficulties that this entails are those having to do with the different shapes that formatives assume in different environments. In other words, the principal difficulties of morphological analysis are in fact phonological or graphological. The inventor of two-level morphology, Kimmo Koskenniemi, is fact provided a finite-state account not just of morphophonemics (or morphographemics), but also of morphotactics. He took it that the allowable set of words simply constituted a regular set of morheme sequences. This is probably the more controversial part of his proposal, but it is also the less technically elaborate, and therefore the one that has attracted less attention. As a result, the term "two-level morphology" has come to be commonly accepted as applying to any system of word recognition that involves two-level, finite-state, phonology or graphology. The approach to nonconcatenative morphology to be outlined in this paper will provide a more unified treatment of morphophonemics and morphotactics than has been usual

I shall attempt to show how a two-level account might be given of nonconcatenative morphological phenomena, particularly those exhibited in the Semitic languages. The approach I intend to take is inspired, not only by finite-state morphology, broadly construed, but equally by autosegmental phonology as proposed by Goldsmith (1979) and the autosegmental morphology of McCarthy (1979). All the data that I have used in this work is taken from McCarthy (1979) and my debt to him will be clear throughout.

forms that can be constructed on the basis of each of the stems shown. However, there is every reason to suppose that, though longer and greatly more complex in detail, that enterprise would not require essentially different mechanisms from the ones I shall describe.

The overall principles on which the material in Table I is organized are clear from a fairly cursory inspection. Each form contains the letters "ktb" somewhere in it. This is the root of the verb meaning "write". By replacing these three letters with other appropriately chosen

	Perfective		*Imperfective*		*Participle*	
	Active	Passive	Active	Passive	Active	Passive
I	katab	kutib	aktub	uktab	kaatib	maktuub
II	kattab	kuttib	ukattib	ukattab	mukattib	mukattab
III	kaatab	kuutib	ukaatib	ukaatab	mukaatib	mukaatab
IV	?aktab	?uktib	u?aktib	u?aktab	mu?aktib	mu?aktab
V	takattab	tukuttib	atakattab	utakattab	mutkattib	mutakattab
VI	takaatab	tukuutib	atakaatab	utakaatab	mutakaatib	mutakaatab
VII	nkatab	nkutib	ankatib	unkatab	minkatib	munkatab
VIII	ktatab	ktutib	aktatib	uktatab	muktatib	muktatab
IX	ktabab		aktabib		muktabib	
X	staktab	stuktib	astaktib	ustaktab	mustaktib	mustaktab
XI	ktaabab		aktaabib		muktaabib	
XII	ktawtab		aktawtib		muktawtib	
XIII	ktawwab		aktawwib		muktawwib	
XIV	ktanbab		aktanbib		muktanbib	
XV	ktanbay		aktanbiy		muktanbiy	

Table I

I take it as my task to describe how the members of a paradigm like the one in Table I might be generated and recognized effectively and efficiently, and in such a way as to capture and profit from the principal linguistic generalizations inherent in it. Now this is a slightly artificial problem because the forms given in Table I are not in fact words, but only verb stems. To get the verb forms that would be found in Arabic text, we should have to expand the table very considerably to show the inflected sequences of three consonants, we would obtain corresponding paradigms for other roots. With some notable exceptions, the columns of the table contain stems with the same sequence of vowels. Each of these is known as a *vocalism* and, as the headings of the columns show, these can serve to distinguish perfect from imperfective, active from passive, and the like. Each row of the table is characterized by a particular pattern according to which the vowels and consonants alternate. In other words, it is characteristic of a given row

that the vowel in a particular position is long or short, or that a consonant is simple or geminate, or that material in one syllable is repeated in the following one. McCarthy refers to each of these patterns as a *prosodic template*, a term which I shall take over. Each of them adds a particular semantic component to the basic verb, making it reflexive, causative, or whatever. Our problem, will therefore involve designing an abstract device capable of combining components of these three kinds into a single sequence. Our solution will take the form of a set of one or more finite-state transducers that will work in parallel like those of Koskenniemmi(1983), but on four tapes rather than just two.

There will not be space, in this paper, to give a detailed account, even of all the material in Table I, not to mention problems that would arise if we were to consider the full range of Arabic roots. What I do hope to do, however, is to establish a theoretical framework within which solutions to all of these problems could be developed.

We must presumably expect the transducers we construct to account for the Arabic data to have transition functions from states and quadruples of symbols to states. In other words, we will be able to describe them with transition diagrams whose edges are labeled with a vector of four symbols. When the automaton moves from one state to another, each of the four tapes will advance over the symbol corresponding to it on the transition that sanctions the move.

I shall allow myself some extensions to this basic scheme which will enhance the perspicuity and economy of the formalism without changing its essential character. In particular, these extensions will leave us clearly within the domain of finite-state devices. The extensions have to do with separating the process of reading or writing a symbol on a tape, from advancing the tape to the next position. The quadruples that label the transitions in the transducers we shall be constructing will be elements each consisting of two parts, a symbol, and an instruction concerning the movement of the tape. I shall use the following notation for this. A unadorned symbol will be read in the traditional way, namely, as requiring the tape on which that symbol appears to move to the next position as soon as it has been read or written. If the symbol is shown in brackets, on the other hand, the tape will not advance, and the quadruple specifying the next following transition will therefore clearly have to be one that specifies the same symbol for that tape, since the symbol will still be under the read-write head when that transition is taken. With this convention, it is natural to dispense with the ε symbol in favor of the notation "[]", that is, an unspecified symbol over which the corresponding tape does not advance. A symbol can also be written in braces, in which case the corresponding tape will move if the symbol under the read-write head is the last one on the tape. This is intended to capture the notion of *spreading*, from autosegmental morphology, that is, the principal according to which the last item in a string may be reused when required to fill several positions.

A particular set of quadruples, or *frames*, made up of symbols, with or without brackets or braces, will constitute the *alphabet* of the automata, and the "useful" alphabet must be the same for all the automata because none of them can move from one state to another unless the others make an exactly parallel transition. Not surprisingly, a considerable amount of information about the language is contained just in the constitution of the alphabet. Indeed, a single machine with one state which all transitions both leave and enter will generate a nontrivial subset of the material in Table I. An example of the steps involved in generating a form that depends only minimally on information embodied in a transducer is given in table II.

The eight step are labeled (a) - (h). For each one, a box is shown enclosing the symbols currently under the read-write heads. The tapes move under the heads from the right and then continue to the left. No symbols are shown to the right on the bottom tape, because we are assuming that the operation chronicled in these diagrams is one in which a surface form is being

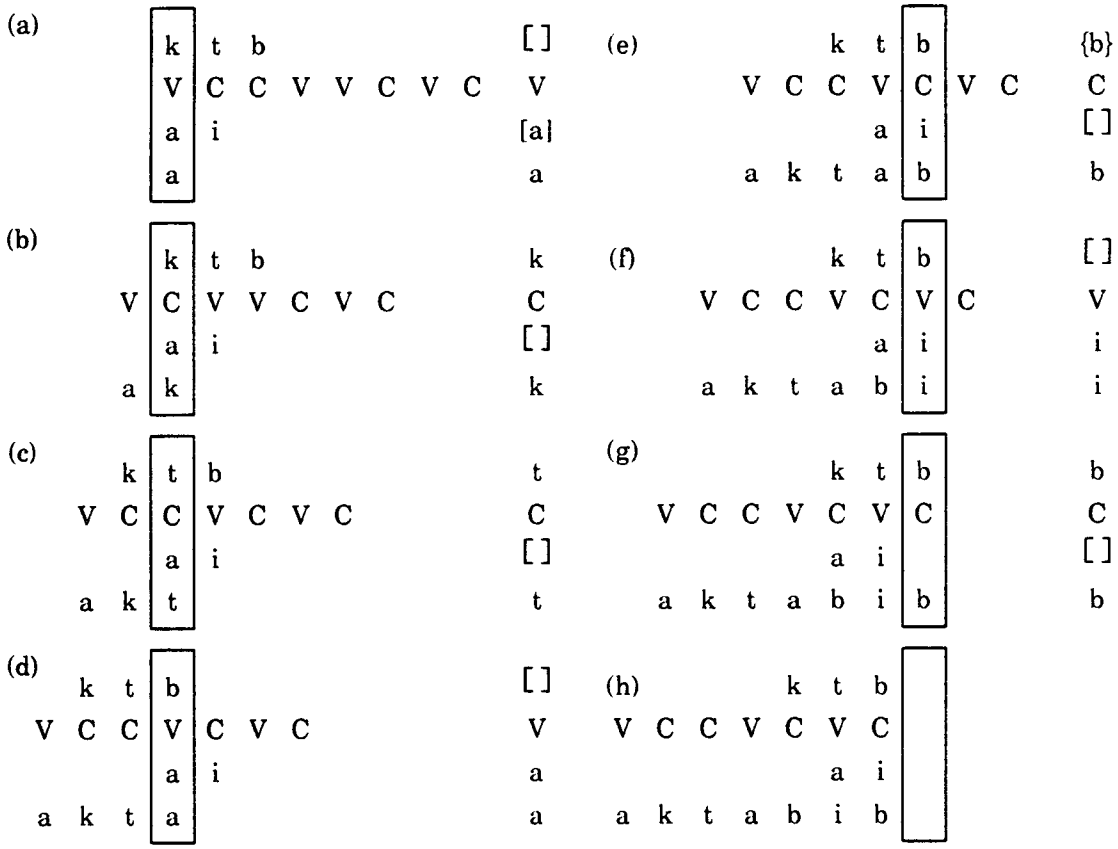

Table II

generated. The bottom tape—the one containing the surface form—is therefore being written and it is for this reason that nothing appears to the right. The other three tapes, in the order shown, contain the root, the prosodic template, and the vocalism. To the right of the tapes, the frame is shown which sanctions the move that will be made to advance from that position to the next. No such frame is given for the last configuration for the obvious reason that this represents the end of the process.

The move from (a) to (b) is sanctioned by a frame in which the root consonant is ignored. There must be a "V" on the template tape and an "a" in the current position of the vocalism. However, the vocalism tape will not move when the automata move to their next states. Finally, there will be an "a" on the tape containing the surface form. In summary, given that the prosodic template calls for a vowel, the next vowel in the vocalism has been copied to the surface. Nondeterministically, the device predicts that this same contribution from the vocalism will also be required to fill a later position.

The move from (b) to (c) is sanctioned by a frame in which the vocalism is ignored. The template requires a consonant and the frame accordingly specifies the same consonant on both the root and the surface tapes, advancing both of them. A parallel move, differing only in the identity of the consonant, is made from (c) to (d). The move from (d) to (e) is similar to that from (a) to (b) except that, this time, the vocalism tape does advance. The nondeterministic prediction that is being made in this case is that there will be no further slots for the "a" to fill. Just what it is that makes this the "right" move is a matter to which we shall return. The move from (e) to (f)

differs from the previous two moves over root consonants in that the "b" is being "spread". In other words, the root tape does not move, and this possibility is allowed on the specific grounds that it is the last symbol on the tape. Once again, the automata are making a nondeterministic decision, this time that there will be another consonant called for later by the prosodic template and which it will be possible to fill only if this last entry on the root tape does not move away. The moves from (f) to (g) and from (g) to (h) are like those from (d) to (e) and (b) to (c) respectively.

Just what is the force of the remark, made from time to time in this commentary, that a certain move is made *nondeterministically*? These are all situations in which some other move was, in fact, open to the transducers but where the one displayed was carefully chosen to be the one that would lead to the correct result. Suppose that, instead of leaving the root tape stationary in the move from (e) to (f), it had been allowed to advance using a frame parallel to the one used in the moves from (b) to (c) and (c) to (d), a frame which it is only reasonable to assume must exist for all consonants, including "b". The move from (f) to (g) could still have been made in the same way, but this would have led to a configuration in which a consonant was required by the prosodic template, but none was available from the root. A derivation cannot be allowed to count as complete until all tapes are exhausted, so the automata would have reached an impasse. We must assume that, when this happens, the automata are able to return to a preceding situation in which an essentially arbitrarily choice was made, and try a different alternative. Indeed, we must assume that a general backtracking strategy is in effect, which ensures that all allowable sequences of choices are explored.

Now consider the nondeterministic choice that was made in the move from (a) to (b), as contrasted with the one made under essentially indistinguishable circumstances from (d) to (e). If the vocalism tape had advanced in the first of these situations, but not in the second, we should presumably have been able to generate the putative form "aktibib", which does not exist. This can be excluded only if we assume that there is a transducer that disallows this sequence of events, or if the frames available for "i" are not the same as those for "a". We are, in fact, making the latter assumption, on the grounds that the vowel "i" occurs only in the final position of Arabic verb stems.

Consider, now, the forms in rows II and V of table I. In each of these, the middle consonant of the root is geminate in the surface. This is not a result of spreading as we have described it, because spreading only occurs with the last consonant of a root. If the prosodic template for row II is "CVCCVC", how is that we do not get forms like "katbab" and "kutbib" beside the ones shown? This is a problem that is overcome in McCarthy's autosegmental account only at considerable cost. Indeed, is is a deficiency of that formalism that the only mechanisms available in it to account for gemination are as complex as they are, given how common the phenomenon is.

Within the framework proposed here, gemination is provided for in a very natural way. Consider the following pair of frame schemata, in which *c* is and arbitrary consonant:

c	[*c*]
C	G
[]	[]
c	*c*

The first of these is the one that was used for the consonants in the above example except in the situation for the first occurrence of "b", where is was being spread into the final two consonantal positions of the form. The second frame differs from this is two respects. First, the prosodic template contains the hitherto unused symbol "G", for "geminate", and second, the root tape is not advanced. Suppose, now, that the the prosodic template for forms like "kattab" is not "CVCCVC", but "CVGCVC". It will be possible to discharge the "G" only if the root template does not advance, so that the following "C" in the template can only cause the same consonant to be inserted into the word a second time. The sequence "GC" in a prosodic template is therefore an idiom for consonant gemination.

Needless to say, McCarthy's work, on which this paper is based, is not interesting simply for the fact that he is able to achieve an adequate description of the data in table I, but also for the claims he makes about the way that account extends to a wider class of phenomena, thus achieving a measure of explanatory power. In particular, he claims that it extends to roots with two and four consonants. Consider, in particular, the following sets of forms:

ktanbab	dhanraj
kattab	dahraj
takattab	tadahraj

Those in the second column are based on the root /dhrj/. In the first column are the corresponding forms of /ktb/. The similarity in the sets of corresponding forms is unmistakable. They exhibit the same patterns of consonants and vowels, differing only in that, whereas some consonant appears twice in the forms in column one, the consonantal slots are all occupied by different segments in the forms on the right. For these purposes, the "n" of the first pair of forms should be ignored since it is contributed by the prosodic template, and not by the root.

consonantal slot in the prosodic template only in the case of the shorter form. The structure of the second and third forms is equally straighforward, but it is less easy to see how our machinery could account for them. Once again, the template calls for four root consonants and, where only three are provided, one must do double duty. But in this case, the effect is achieved through gemination rather than spreading so that the gemination mechanism just outlined is presumably in play. That mechanism makes no provision for gemination to be invoked only when needed to fill slots in the prosodic template that would otherwise remain empty. If the mechanism were as just described, and the triliteral forms were "CVGCVC" and "tVCVGCVC" respectively, then the quadriliteral forms would have to be generated on a different base.

It is in cases like this, of which there in fact many, that the finite-state transducers play a substantive role. What is required in this case is a transducer that allows the root tape to remain stationary while the template tape moves over a "G", provided no spreading will be allowed to occur later to fill consonantal slots that would

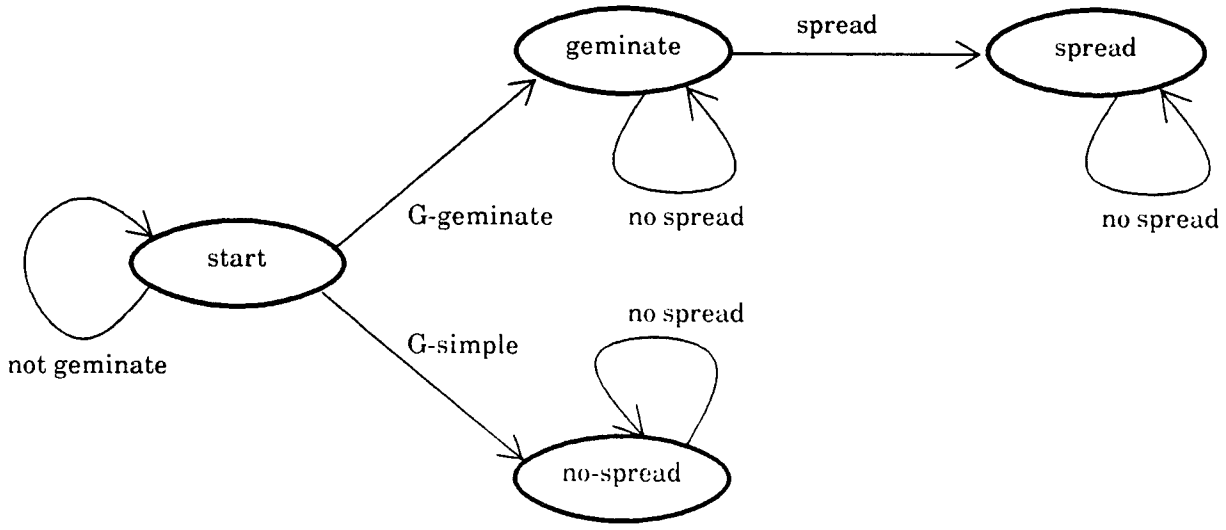

Fig. 1

Given a triliteral and a quadriliteral root, the first pair are exactly as one would expect—the final root consonant is spread to fill the final otherwise be unclaimed. If extra consonants are required, then the first priority must be to let them occupy the slots marked with a "G" in the

template. Fig. 1 shows a schema for the transition diagram of a transducer that has this effect. I call it a "schema" only because each of the edges shown does duty for a number of actual transitions. The machine begins in the "start" state and continues to return there so long as no frame is encountered involving a "G" on the template tape. A "G" transition causes a nondeterministic choice. If the root tape moves at the same time as the "G" is scanned, the transducer goes into its "no-spread" state, to which it continues to return so long as every move over a "C" on the prosodic tape is accompanied by a move over a consonant on the root tape. In other words, it must be possible to complete the process without spreading consonants. The other alternative is that the transducer should enter the "geminate" state over a transition over a "G" in the template with the root tape remaining stationary. The transitions at the "geminate" state allow both spreading and nonspreading transitions. In summary, spreading can occur only if the transducer never leaves the "start" state and there is no "G" in the template, or there is a "G" on the template which does not trigger gemination. A "G" can fail to trigger gemination only when the root contains enough consonants to fill all the requirements that the template makes for them.

One quadriliteral case remains to be accounted for, namely the following:

ktaabab dharjaj

According to the strategy just elaborated, we should have expected the quadriliteral form to have been "dhaaraj". But, apparently this form contains a slot that is used for vowel lengthening with triliteral roots, and as consonantal position for quadriliterals. We must therefore presumably take it that the prosodic template for this form is something like "CCVXCVC" where "X" is a segment, but not specified as either vocalic or consonantal. This much is in line with the proposal that McCarthy himself makes. The question is, when should be filled by a vowel, and when by a consonant? The data in Table I is, of course, insufficient to answer question, but a plausible answer that strongly suggests itself is that the "X" slot prefers a consonantal filler *except* where that would result in gemination. If this is true, then it is another case where the notion of gemination, though not actually exemplified in the form, plays a central role. Supposing that the analysis is correct, the next question is, how is it to be implemented. The most appealing answer would be to make "X" the exact obverse of "G", when filled with a consonant. In other words, when a root consonant fills such a slot, the root tape must advance so that the same consonant will no longer be available to fill the next position. The possibility that the next root consonant would simply be a repetition of the current one would be excluded if we were to take over from autosegmental phonology and morphology, some version of th *Obligatory Contour Principle* (*OCP*) (Goldsmith, 1979) which disallows repeated segments except in the prosodic template and in the surface string. McCarthy points out the roots like /smm/, which appear to violate the OCP can invariably be reanalyzed as biliteral roots like /sm/ and, if this is done, our analysis, like his, goes through.

The OCP does seem likely to cause some trouble when we come to treat one of the principal remaining problems, namely that of the forms in row I of table I. It turns out that the vowel that appears in the second syllable of these forms is not provided by the vocalism, but by the root. The vowel that appears in the perfect is generally different from the one that appears in the imperfect, and four different pairs are possible. The pair that is used with a given root is an idiosyncratic property of that root. One possibility is, therefore, that we treat the traditional triliteral roots as consisting not simply of three consonants, but as three consonants with a vowel intervening between the second and third, for a total of four segments. This flies in the face of traditional wisdom. It also runs counter to one of the motivating intuitions of autosegmental phonology which would have it that particular phonological features can be represented on at most one lexical tier, or tape. The intuition is that these tiers or tapes each contain a record or a particular kind of

articulatory gesture; from the hearer's point of view, it is as though they contained a record of the signal received from a receptor that was attuned only to certain features. If we wish to maintain this model, there are presumably two alternatives open to us. Both involve assuming that roots are represented on at least two tapes in parallel, with the consonants separate from the vowel.

According to one alternative, the root vowel would be written on the same tape as the vocalism; according to the other, it would be on a tape of its own. Unfortunately, neither alternative makes for a particularly happy solution. No problem arises from the proposal that a given morpheme should, in general, be represented on more than one lexical tape. However, the idea that the vocalic material associated with a root should appear on a special tape, reserved for it alone, breaks the clean lines of the system as so far presented in two ways. First, it spearates material onto two tapes, specifically the new one and the vocalism, on purely lexical grounds, having nothing to do with their phonetic or phonological constitution, and this runs counter to the idea of tapes as records of activity on phonetically specialized receptors. It is also at least slightly troublesome in that that newly introduced tape fills no function except in the generation of the first row of the table. Neither of these arguments is conclusive, and they could diminish considerably in force as a wider range of data was considered.

Representing the vocalic contribution of the root on the same tape as the vacalism would avoid both of these objections, but would require that vocalic contribution to be recorded either before or after the vocalism itself. Since the root vowel affects the latter part of the root, it seems reasonable that it should be positioned to the right. Notice, however, that this is the only instance in which we have had to make any assumptions about the relative ordering of the morphemes that contribute to a stem. Once again, it may be possible to assemble further evidence reflecting on some such ordering, but I do not see it in these data.

It is only right that I should point out the difficulty of accounting satisfactorily for the vocalic contribution of verbal roots. It is only right that I should also point out that the autosegmental solution fares no better on this score, resorting, as it must, to rules that access essentially non-phonological properties of the morphemes involved. By insisting that what I have called the *spelling* of a morpheme should by, by definition, be its only contribution to phonological processes, I have cut myself off from any such *deus ex machina*.

Linguists in general, and computational linguists in particular, do well to employ finite-state devices wherever possible. They are theoretically appealing because they are computational weak and best understood from a mathematical point of view. They are computationally appealing because they make for simple, elegant, and highly efficient implementaions. In this paper, I hope I have shown how they can be applied to a problem in nonconcatenative morphology which seems initially to require heavier machinary.

REFERENCES

Goldsmith, J A. (1979). *Autosegmental Phonology*. New York; Garland Publishing Inc.

Kay, M and R. M. Kaplan (in preparation). *Phonological Rules and Finite-State Transducers*.

Koskenniemi, K (1983). *Two-Level Morphology: A General Computational Model for Word-Form Recognition and Production*. Doctoral Dissertation, University of Helsinki.

Leben, W (1973). Suprasegmental Phonology. Doctoral Dissertation, MIT, Cambridge Massachussetts.

McCarthy, J J. (1979). *Formal problems in Semitic Phonology and Morphology*. Doctoral Dissertation, MIT, Cambridge Massachussetts.

McCarthy, J J. (1981). "A Prosodic Tehory of Nonconcatenative Morphology". *Linguistic Inquiry*, 12.3.