Computational Locality and Autosegmental Processes

Jane Chandlee (Haverford College) and Adam Jardine (Rutgers University)

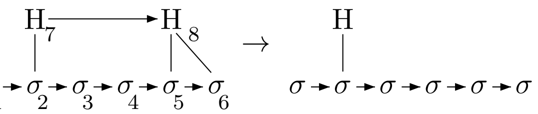
It has long been claimed that autosegmental representations (ASRs; Goldsmith 1976, Clements 1976) can capture non-local phenomena in a local way (McCarthy, 1982; Odden, 1994). As McCarthy (1982) describes, "The apparent action-at-a-distance in vowel harmony—its nonlocal effect—becomes strictly local when viewed as an autosegmental phenomenon."

An example from the domain of tone is Arusa deletion (Levergood 1987; Odden, 1994): a phrase-final high (H) tone is deleted following another H tone, no matter how many syllables intervene (1). (The indices in the input ASR will be explained momentarily.)

(1) Arusa (Levergood 1987; Odden 1994)

a. síd́áy 'good'
b. enkér siday 'good ewe'
c. olórika siday 'good chair'

(2)



/olórika síd́áy/ → [olórika siday]

In the ASR in (2) representing example (1c), the second high tone of /olórika síd́áy/ is adjacent to the first, made explicit with an arrow representing the *successor* ordering relation on each tier. This contrasts with how the process would be viewed as operating over a string of syllables, as the targeted syllables are *not* adjacent to the triggering syllable. Thus, the process is only 'local' when viewed over ASRs.

In this paper we explore ASRs' ability to 'localize' phenomena using a computationally well-defined property of locality. We will analyze a range of tone phenomena and show that while processes like in Arusa are indeed local in this computational sense, not all ASR processes are. We thus obtain a more rigorously-defined notion of when ASRs make processes local.

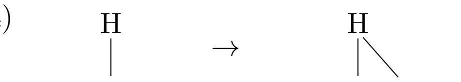
The formalism we use is first-order (FO) logical formulae with variables that refer to the positions in either the segmental or tone string. For example, in the input graph of (2), the mora unit positions are numbered 1-6 starting from the left, and the two tone unit positions are 7 and 8. We can then define a formula to pick out certain positions based on their label (σ, H, L, etc.), their inter-tier associations (using a), and what comes before (using p or predecessor) or after (using s or successor) them. For example, the formula $\sigma(x) \wedge a(x, y) \wedge H(y)$ says a position is labeled σ and is associated to a position that is itself labeled H. This formula evaluates to true for positions 2, 5, and 6.

An input-output map is achieved as follows. Using these formulae to access information about the input graph positions, we 'build' the corresponding output graph by defining the needed positions, labels, and relations (predecessor, successor, association) (see Engelfriet and Hoogeboom, 2001, for details). For example, we can model Cicembe bounded tone spreading, shown in (3), as follows.

(3) Cicembe (Meyers, 1997)

a. tu-la-kak-a 'we tie up'
b. bá̲-lá-kak-a 'they tie up'
c. tu-la-bá̲-kák-a 'we tie them up'

(4)



The formula in (5) defines an output position x associated to an output position y if either 1) the input correspondent of x is associated to the input correspondent of y, or 2) the predecessor of the input correspondent of x is associated to the input correspondent of y.

(5) Output formula for Cicembe

$$a_O(x, y) \stackrel{\text{def}}{=} a_I(x, y) \vee a_I(p(x), y)$$

(iff x and y are associated in the input, or $p(x)$ and y are)

Additional formulae are needed to specify the output position labels and the successor/predecessor relations (which in this example remain the same from input to output). We omit these here for clarity and to emphasize that this process primarily involves the association relation.

As another example, the formula in (6) is a partial analysis of Arusa tone deletion shown above in (1) and (2): an output position is labeled H if it was labeled H in the input AND its predecessor in the input is *not* labeled H. (An output position with no label is then deleted.)

(6) Output formula for Arusa

$$\mathrm{H}_O(x, y) \overset{\text{def}}{=} \mathrm{H}_I(x) \wedge \neg(\mathrm{H}_I(p(x)))$$

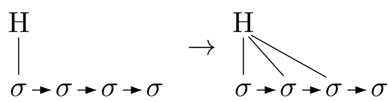
(iff x is an H in the input, and its predecessor is not *an H)*

Importantly, all of these formulae are *quantifier-free* (QF) FO formula, a limited subset of FO logic. QF FO logic been shown to correspond in string representations to a computationally well-defined notion of local processes (Chandlee and Lindell in prep; Chandlee 2014); we here extend it to ASR graphs. The ability to do without quantifiers stems from the fact that all the needed information can be found within a bounded window of material surrounding a given position. The use of the predecessor and successor relations is thus sufficient.

To see this contrast between QF and non-QF, consider the example of spreading in Shambaa, shown in (7) and (8). A prefix high tone spreads to following syllables up until the final vowel.

(7) Shambaa (Odden, 1982)

a.	ku-ʃunt^h-a	'to wash'
b.	ku-tʃí-ʃúnt^h-a	'to wash it'
c.	ku-ɣoʃo-a-ɣoʃo-a	'to do repeatedly'
d.	ku-tʃí-ɣóʃó-á-ɣóʃó-a	'to do it repeatedly'

(8)


Because this spreading is unbounded, to determine whether an output position is associated to the high tone requires non-local information: the input H-tone may be associated to a syllable any number of positions to the left.

Another non-QF example is found in Shona (Odden 1986; Meyers 1987, 1997). As shown in (9), H→L lowering takes place on alternating syllables.

(9) Shona (tonal tier indicated underneath)

a.	/né-hóvé/	→	[né-hòvè]	'with a fish'
	H H	→	H L	
b.	/né-é-hóvé/	→	[né-è-hóvé]	'with-of-fish'
	H H H	→	H L H	
c.	/né-é-é-hóvé/	→	[né-è-é-hòvè]	'like-with-of-fish'
	H H H H	→	H L H L	

In terms of the input alone, a QF formula cannot pick out which high tones will lower. Following a syllable associated to a high tone is not a sufficient condition, since this is true for both the second and third syllables of (9b), yet only the second lowers.

This criterion—defining a QF formula that identifies targets of the process in terms of information present in the input—provides a strict notion of computational locality. This in turn provides a more rigorous, independent notion of what it means for a process to be local over ASRs. Future work can draw on other logical techniques (e.g., using precedence instead of successor; Heinz 2010) to characterize the non-local processes.

Select references: • Chandlee, J. and Lindell, S. (in prep.). A logical characterization of strictly local functions. In Heinz, J., editor, *Doing Computational Phonology*. OUP. • Engelfriet, J. and Hoogeboom, H. J. (2001). MSO definable string transductions and two-way finite-state transducers. *ACM Transations on Computational Logic*, 2:216–254. • McCarthy, J. (1982). Nonlinear phonology: An overview. *GLOW Newsletter*, 50. • Meyers, S. (1997).

OCP effects in Optimality Theory. *NLLT*, 15(4):847–892. • Odden, D. (1994). Adjacency parameters in phonology. *Language*, 70(2):289–330.