



# Input and output locality and representation

**JANE CHANDLEE** 

**ADAM JARDINE** 

*\*Author affiliations can be found in the back matter of this article*

**RESEARCH**

**]u[ubiquity press**

## Abstract

Using a rigorous, computational notion of locality, this paper evaluates one of the central motivations for autosegmental representations (ARs)—that they reduce long-distance processes to local ones. We analyze a variety of tone processes using two computational notions of locality: input strict locality as defined by Chandlee (2014) and Chandlee & Jardine (2019a) and a corresponding notion of output strict locality we call *recursive strict locality*. The results of our survey add to our understanding of the typology of tone patterns and the role of ARs in two key ways. First, they indicate that both input and output locality play a role in a comprehensive theory of tone. Second, they reveal the various mechanisms by which ARs render long-distance patterns local by disentangling the various properties these representations combine. The larger contribution then is a more detailed and nuanced exploration of the interaction of representation, locality, and computational complexity in the domain of tonal phonology.

**CORRESPONDING AUTHOR:**

**Jane Chandlee**

Haverford College, US

[jchandlee@haverford.edu](mailto:jchandlee@haverford.edu)

---

**KEYWORDS:**

tone processes; autosegmental representations; computational analysis; locality

**TO CITE THIS ARTICLE:**

Chandlee, Jane and Adam Jardine. 2021. Input and output locality and representation. *Glossa: a journal of general linguistics* 6(1): 43. 1–27. DOI: <https://doi.org/10.5334/gjgl.1423>

Locality has long played a role in phonological theory as a restriction on possible grammars, with various proposals put forth over the years for what it means to be ‘local’. Preserving locality was in fact one of the major motivating factors behind the use of feature-geometric, autosegmental representations (ARs; Williams 1976; Goldsmith 1976; Clements 1977). As Odden (1994: 289) states, the advantage of ARs is that they “[make] it possible to view apparently long-distance rules as rules operating between segments which are adjacent at a specified level, even though the segments are not adjacent at all levels of representation”. However, since the advent of Optimality Theory (OT; Prince & Smolensky 1993), the utility of nonlinear representations has explicitly come into question (Leben 2006; Hyman 2014a; Shih & Inkelas 2019).

At the same time, recent work in theoretical computational linguistics has made it possible to rigorously study representations and locality. There are many ways in which locality can be conceptualized, but theoretical computer science provides one grounded in principles of computation that can be readily applied to natural language in general and phonology in particular (Rogers & Pullum 2011; Chandlee & Heinz 2018). Briefly, a local computation is one which operates over some contiguous sequence of  $k$  positions in a string, as exemplified in (1).

$$(1) \quad \boxed{a_1} \boxed{a_2} \dots \boxed{\tau_1} \boxed{\tau_2} \dots \boxed{\tau_k} \dots \boxed{a_n}$$

For example, a \*CLASH constraint against adjacent H-toned syllables (Zoll 2003), as in (2a), is local in this sense because it evaluates contiguous sequences of length 2. The string in (2b) violates (2a) because it contains the forbidden sequence, whereas (2c) does not.

- (2) a. \*óó  
 b. σσσóóσ  
 c. óóσσóσ

However, it is common for phonological elements to interact at a distance, and this is particularly prevalent in tone (Yip 2002; Kisseberth & Odden 2003; Hyman 2009). For example, in languages like Somali or Arusa (see more below), a sequence of H tones is banned in certain domains, regardless of the distance between them (see Hyman 2009 for a survey of examples). To use the above example, both (2b) and (2c) would be ungrammatical in Somali. A long-distance Obligatory Contour Principle-style constraint could ban any sequence of H tones regardless of adjacency. However, this is clearly not local in the above-defined computational conception—at least over surface strings. The challenge is then to capture these long-distance processes, but in a restrictive way. This has been the goal of using ARs to reconceive of long-distance processes as local ones.

Indeed, the computational study of phonotactics has adapted autosegmental concepts to do exactly this. Recent work has shown the utility of tier projections (Heinz et al. 2011; McMullin & Hansson 2016) and ARs (Jardine 2017; 2019; 2020) in capturing long-distance patterns with grammars that are local in this way. For example, the constraint in (2a) can be reformulated over ARs as a constraint over two H tones adjacent on the tonal tier, as in (3a); this is violated both by adjacent H-toned tone bearing units (TBUs), as in (3b), or H tones that are associated to non-adjacent TBUs (3c). This is because both contain the forbidden HH sequence *on the tonal tier*. In this way, ARs capture long-distance phonotactic patterns with local constraints.

- (3) a. \*HH  
 b.  $\begin{array}{cccccc} & & & & \text{H} & \text{H} \\ & & & & | & | \\ \sigma & \sigma & \sigma & \sigma & \sigma & \sigma \end{array}$   
 c.  $\begin{array}{cccccc} \text{H} & & & & \text{H} & \\ | & & & & | & \\ \sigma & \sigma & \sigma & \sigma & \sigma & \sigma \end{array}$

A largely unanswered question, however, is how these results can be extended to a local characterization of phonological *processes*, which can be studied computationally by modeling them as functions. Tonal processes are a particularly apt testing ground for this, as tone exhibits a vast range of both local and non-local processes (Yip 2002; Hyman 2009). As a first

step, Chandlee & Jardine (2019a) propose the *autosegmental input strictly local* (A-ISL) class of functions that transform underlying ARs to surface ARs, based on Chandlee (2014)'s *input strictly local* (ISL) functions for strings. Both ISL and A-ISL formalize a notion of locality in which any change is based on information in the *input* that is within some fixed distance of that change. The difference is only that with ISL that input is a string and with A-ISL that input is an AR.

This paper expands on the results of Chandlee & Jardine (2019a) to lay the foundation for a more comprehensive computational theory of tone. In particular, we first introduce a corresponding output-based notion of locality over ARs that parallels A-ISL. We call this class of functions the *autosegmental recursive strictly local* (A-RSL) functions and define them using a logical characterization of output locality inspired by the output strictly local (OSL) functions for strings defined by Chandlee et al. (2015a). We show how this class of functions captures those tone processes that cannot be captured by ISL or A-ISL because their computation is necessarily output-oriented. From there we establish a four-way distinction of locality in tone processes, along the dimensions of strings versus ARs and input- versus output-based computation: 1) input-local over strings, 2) input-local over ARs, 3) output-local over strings, and 4) output-local over ARs.

Next we apply these four categories to a series of tone processes, which includes bounded and unbounded tone spread and tone shift, as well as several variants of Meeussen's rule, establishing for each which categories it does and does not belong to. The results show that all four categories have utility in accommodating the range of attested tone processes (i.e., no one category subsumes the other). At the same time, it is not evident from this survey alone that all four categories play *distinct* roles in the phonological grammar, as some processes can be equally modeled with multiple categories, and the question of which category offers the 'best' characterization is a matter of debate.

These results have larger implications for our understanding of tone in particular and the phonological grammar more broadly in the following two ways. One, we are constructing a computational theory of tone, one grounded in locality and its interaction with representation. The analysis of tone processes in isolation suggests that in such a theory the phonological grammar has more than one option for asserting locality.<sup>1</sup> However, identifying which of the available options are both necessary and sufficient requires a broader exploration into the larger typology of tone processes and their interaction with other aspects of the phonological grammar. Given that, our computational theory of tone is necessarily incomplete for now, but our results point to the hypotheses and predictions that will get us there.

Two, regardless of one's theoretical orientation, the computational framework provided here for investigating tone offers a means of comparing competing analyses of a process that differ in representational assumptions or classification as local or non-local. In other words, it offers a way of identifying the computational cost of differing assumptions as well as the (non) exceptionality of a given process in relation to others. We will see examples of this in our survey.

## 1.1 Overview of the proposal

To briefly demonstrate the computational framework used in this paper, consider the process of unbounded tone shift in *Zigula* (Kenstowicz & Kisseberth 1990), in which an underlying H tone shifts to the penultimate TBU in the word regardless of how many TBUs intervene between those two positions.

(4)	<i>Zigula</i> (Niger-Congo; Kenstowicz & Kisseberth 1990)		
a.	/ku-songoloz-a/	[ku-songoloz-a]	'to avoid'
b.	/á-songoloz-a/	[a-songolóz-a]	'he/she is avoiding'
c.	/ku-lómbez-a/	[ku-lombéz-a]	'to ask'
d.	/ku-lómbez-ez-an-a/	[ku-lombez-ez-án-a]	'to ask for each other'

<sup>1</sup> We analyze processes in isolation in order to focus on the interaction of locality and representation, but this computational framework is not limited to single-process analyses. In Appendix A we demonstrate its broader utility with a unified analysis of multiple related processes in Shona.

In (5) we show this shift for example (4b).



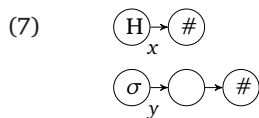
We here present a brief analysis of Zigula tone shift as an A-ISL function, with a more detailed presentation of the framework to follow in the next section. Following Chandlee & Jardine (2019a), we define A-ISL using *quantifier-free (QF) logical transductions*. QF transductions reference input representations using a predicate logic in which variables range over elements (or *positions*) in the representation—but quantifiers are disallowed. For example, we can establish which tones and TBUs should be associated to each other in the output structure using the logical formula in (6). This formula defines the associations that hold between tones and TBUs in the output structure.

(6) 
$$A_o(x, y) \stackrel{\text{def}}{=} \text{H}(x) \wedge \#(s(x)) \wedge \sigma(y) \wedge \#(s(s(y)))$$

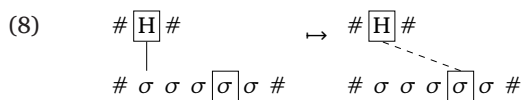
The conditions for association specified by this formula are broken down below. (Note that *s* is the *successor* function that establishes the linear order of positions.)

$A_o(x, y) \stackrel{\text{def}}{=}$	an output position <i>y</i> should be associated to an output position <i>x</i> iff
$\text{H}(x) \wedge$	<i>x</i> is a high tone, and ...
$\sigma(y) \wedge$	<i>y</i> is a TBU, and ...
$\#(s(x)) \wedge$	the tone <i>x</i> is one position from the end of the tonal tier, and ...
$\#(s(s(y)))$	the TBU <i>y</i> is two positions away from the end of the TBU tier

This configuration is depicted visually as a graph in (7), with the positions of the structure depicted as nodes (circles), and the successor function depicted as edges (arrows) between the nodes. The target nodes *x* and *y* that will be associated are also labeled.



The right-hand side of a formula such as (6) then identifies those positions in the input structure that satisfy certain conditions. The left-hand side asserts that the *output correspondents* of such positions are associated. This is shown in the examples in (8), in which boxes highlight the positions that satisfy the formula and their output correspondents.



The significance of the formula being quantifier-free (QF) is that global information that would require scanning the entire input (e.g., ‘there exists a position such that...’) is not available. Rather, the formula is limited to examining a fixed window of material around a particular position. In this way the transduction is required to operate in a local manner. As we show, the fact that A-ISL captures processes that would otherwise not be amenable to a local analysis confirms, from a formal perspective, that the power of ARs comes from the asynchronicity of distinct tiers and the manipulation of the association relation between them.

However, as already noted, the A-ISL class cannot capture processes that require *output* locality, or reference to the output structure in the right-hand side of the formula. To address this gap, we will propose a corresponding class of functions we call A-RSL (autosegmental recursive strictly local).<sup>2</sup> Collectively our results indicate that preserving locality in the domain of tone processes requires both options for representation (strings and ARs)<sup>3</sup> and both types of locality (input and output).

<sup>2</sup> We have opted to not call this class A-OSL because we do not yet have a mathematical proof that it subsumes the existing class of OSL functions.

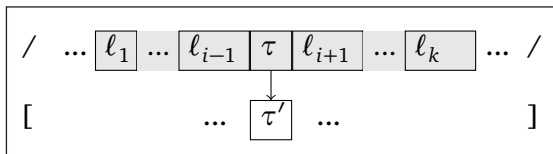
<sup>3</sup> An anonymous reviewer points out that this is also true in some cases in syntax, where some constraints can be simplified when stated over strings instead of trees. For example, the *that*-trace effect can be reduced to a simple constraint against the substring *that-t*, but over trees it becomes less straightforward. We thank the reviewer for pointing out this interesting connection between the two domains.

The remainder of the paper is as follows. §2 details the use of logical transductions to model phonological processes and defines the two types of locality, input locality in §2.1 and output locality in §2.2. In §3 we use this framework in analyses of several types of tone processes, including bounded and unbounded tone shift and tone spread in §3.1 and variants of Meeussens’s rule in §3.2. In §4 we compare these computational notions of locality to previous ones in theoretical phonology, and also discuss the implications of our results for a more comprehensive computational theory of tone processes. §5 then concludes.

## 2 Input and output locality

### 2.1 Input locality

The first function class we will make use of in our analyses is the input strictly local (ISL) functions, which are functions that determine an output string for a given input string based only on input substrings of bounded length (Chandlee 2014; Chandlee et al. 2014).<sup>4</sup> As schematized in *Figure 1*, in an ISL function each element of an input string contributes some (possibly empty) substring to the output, and the content of that substring is based only on the input element itself and the  $k-1$  surrounding input elements. The integer  $k$  then serves as an upper bound on how much of the input string can be used to determine the output.



**Figure 1** Computing an ISL function. Here,  $\tau$  is an arbitrary input element;  $\tau'$  is its output. The shaded portion of the input is the information available to the function when determining  $\tau'$ .

The property of being ISL then provides a precise and computational notion of what it means for a phonological process to be local. One goal for this paper is to assess how well this formal conception of locality aligns with the sense in which autosegmental representations enable a local treatment of otherwise non-local phenomena. In pursuit of this goal we will use logical characterizations of functions, which enable a more direct comparison among different representations (in our case strings versus ARs) compared to previous automata-theoretic characterizations.

Consider a process in which a tone shifts some fixed number of TBUs away from its underlying position. For example, in Rimi, a tone shifts one TBU to the right, as shown in (9). For convenience these examples are rewritten as strings of TBUs in (10). (More will be said about this representational choice in §3.)

- (9) *Rimi* (Niger-Congo; Schadeberg 1979; Myers 1997)
- |    |             |             |               |
|----|-------------|-------------|---------------|
| a. | /u-hang-a/  | [u-hang-a]  | ‘to meet’     |
| b. | /u-pùm-a/   | [u-pùm-á]   | ‘to go away’  |
| c. | /mu-ntu/    | [mu-ntu]    | ‘person’      |
| d. | /rá-mu-ntu/ | [ra-mú-ntu] | ‘of a person’ |
| e. | /u-huvi-ì/  | [u-huvi-ì]  | ‘belief’      |
| f. | /mu-tém-ì/  | [mu-tem-í]  | ‘chief’       |

- (10)
- |    |   |
|----|---|
| a. | $\sigma\sigma\sigma \mapsto \sigma\sigma\sigma$                 |
| b. | $\sigma\acute{\sigma}\sigma \mapsto \sigma\sigma\acute{\sigma}$ |
| c. | $\sigma\sigma \mapsto \sigma\sigma$                             |
| d. | $\acute{\sigma}\sigma\sigma \mapsto \sigma\acute{\sigma}\sigma$ |
| e. | $\sigma\sigma\sigma \mapsto \sigma\sigma\sigma$                 |
| f. | $\sigma\acute{\sigma}\sigma \mapsto \sigma\sigma\acute{\sigma}$ |

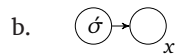
The logical characterization of ISL defines formulas in first order (FO) logic that pick out input elements that meet certain conditions. For example, the formulas  $\acute{\sigma}(x)$  identifies input elements that bear a high tone. We can also use terms like  $p(x)$ , or *predecessor* of  $x$ , and  $s(x)$ , or *successor* of  $x$ , to refer to elements immediately preceding or following  $x$ , respectively.

<sup>4</sup> While Chandlee (2014) defines strictly local functions in the context of phonological processes, local functions themselves have precedent (see Berstel 1982; Vaysse 1986; Lind & Marcus 1995).

A map or transduction from an input to output string involves defining *output predicates* that determine the content of the positions in the output string. Throughout the paper we distinguish these output predicates with a subscript *o*. For example, (11a) specifies the conditions for an output position to bear a high tone.

(11) Bounded shift (ISL) (part 1)

a.  $\acute{\sigma}_o(x) \stackrel{\text{def}}{=} \acute{\sigma}(p(x))$



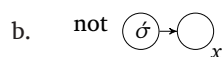
(11a) says that an output element *x* should bear a high tone if its predecessor *p(x)* in the *input* bears a high tone. To illustrate, (12) highlights in boxes those TBUs from (10a), (10b), and (10d) that satisfy (11a) and thus whose output correspondents will be  $\acute{\sigma}$ .

- (12) a.  $\sigma\sigma\sigma$   
 b.  $\sigma\acute{\sigma}\sigma$   
 c.  $\acute{\sigma}\sigma\sigma$

Likewise, (13a) identifies those output elements that are unspecified for tone: namely, those elements whose predecessors in the input do *not* bear a high tone. This formula is satisfied by all of the TBUs in (12) that are not in boxes.

(13) Bounded shift (ISL) (part 2)

a.  $\sigma_o(x) \stackrel{\text{def}}{=} \neg\acute{\sigma}(p(x))$



A complete definition of a logical transduction includes output predicates for all possible output labels (and in the case of ARs also the association relation). In the interest of space our analyses will only provide output predicates for those aspects of the representation that actually change from input to output. All others can be assumed to be defined to retain their value from the input (e.g.,  $a_o(x) \stackrel{\text{def}}{=} a(x)$ ). Finally, note also that as logical definitions define an output structure based on the set of output predicates that evaluate to true, they are guaranteed to define functions, such that no input is mapped to more than one output form (though they can be extended to deal with optionality; see Engelfriet & Hoogeboom 2001).

As noted in the introduction, the transduction is forced to operate in a local way because all of the formulas are *quantifier-free* (QF). The only means of accessing information is through the use of the successor and/or predecessor functions. Calls to these functions can be embedded, such as  $p(p(x))$  to identify the predecessor of a predecessor, but the degree of embedding will be finite.

Chandlee & Jardine (2019b) show that these transductions describe exactly the ISL functions, provided they are *order-preserving*, meaning that a position that precedes/follows another position in the input will also precede/follow that position in the output.<sup>5</sup> We follow this assumption throughout the paper.

It follows then that the ISL functions provide a means of formally distinguishing local from long-distance processes. Consider the example of long-distance consonant agreement found in Kikongo, in which the suffix *-idi* surfaces as *-ini* when it attaches to a stem that contains a nasal consonant.

- (14) *Kikongo* (Niger-Congo; Ao 1991)  
 /tu-nik-idi/ [tunikini] ‘we ground’

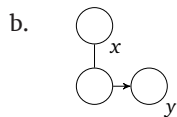
In terms of the FO characterization used in this paper, a formula that determines the nasality of the suffix consonant would require a quantifier to examine the entire stem. The use of embedded predecessor functions is not possible, as there is no upper bound on how many preceding segments must be examined to confirm the presence or absence of a stem nasal.

<sup>5</sup> This might give the impression that we are ruling out metathesis, which is in fact ISL (when local) (see Chandlee 2014). But there is a crucial difference between preserving the order of the *positions* and preserving the order of the *position labels*. We are only ruling out the former. We thank an anonymous reviewer for pointing out this important distinction.

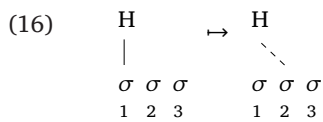
This formal means of distinguishing local from long-distance enables a similarly formal means of investigating how long-distance patterns are made local by ARs. We first must extend ISL to operate over ARs instead of strings. ARs consist of two strings (one of tones and one of TBUs), each with their own predecessor and successor functions.<sup>6</sup> In addition, an association relation *A* defines which tones are linked to which TBUs. The formulas that define transductions over ARs can refer to positions on either string or the associations between them. We assume that the input to these transductions are well-formed ARs. That is, tones and TBUs appear on separate tiers, all associations occur between a tone and a TBU, and all associations obey the no-crossing constraint (for a recent formal overview, see Jardine 2016b).

In the case of Rimi, the transduction involves a change in the association relation, specified by (15a). This formula says that a TBU *y* is associated to a tone *x* in the output if *x* was associated to *y*'s predecessor in the input.

- (15) Bounded shift (A-ISL)  
 a.  $A_o(x, y) \stackrel{\text{def}}{=} A(x, p(y))$



The example in (16) demonstrates how this formula implements the tone shift. In this example,  $A_o(x, y)$  is only true when *x* is interpreted as the H and *y* as syllable 2, and so these two positions (and no others) are associated in the output.<sup>7</sup>

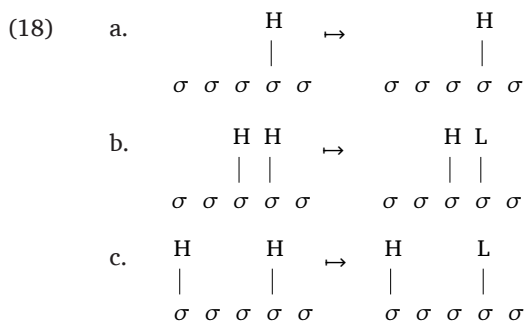


A function such as this one we call A-ISL, to reflect the fact that it operates over ARs instead of strings. Both ISL and A-ISL require the formulas to be QF; they differ only in representations.

As is shown in the analyses below, however, the tier structure of ARs allows for some processes that operate over an arbitrary number of TBUs (and thus are not ISL) to be captured with QF formulas (and thus are A-ISL). To give a hypothetical example, imagine a Meeussen's Rule-type process in which an H becomes L following another H, regardless of the number of intervening TBUs.

- (17) a.  $\sigma\sigma\acute{\sigma}\sigma \mapsto \sigma\sigma\acute{\sigma}\sigma$   
 b.  $\sigma\acute{\sigma}\acute{\sigma}\sigma \mapsto \sigma\acute{\sigma}\acute{\sigma}\sigma$   
 c.  $\acute{\sigma}\sigma\acute{\sigma}\sigma \mapsto \acute{\sigma}\sigma\acute{\sigma}\sigma$

This process is not ISL for the same reason that Kikongo nasal agreement isn't: determining whether or not an underlying /*ó*/ surfaces as [ð] requires a quantifier to examine the entire string, as a triggering *ó* can be arbitrarily far away. However, this process is A-ISL, as the two H tones will be adjacent on the tonal tier, as shown in (18).



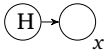
<sup>6</sup> We only consider ARs with two strings, leaving the analysis of ARs with more than two strings to future work.

<sup>7</sup> Throughout the paper, we use dashed association lines to indicate associations that satisfy the output association formula.



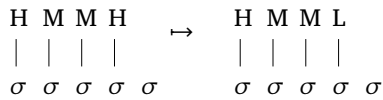
This process is described exactly by the QF definition for an output L given in (19a).

(19) Long-distance Meeussen’s rule (A-ISL)

- a.  $L_o(x) \stackrel{\text{def}}{=} H(p(x))$
- b. 

However, note that this analysis depends on the assumption of a privative H/∅ system. If, for example, there could be any number of intervening M tones, as in (20), the triggering H would no longer be within a fixed distance of the target H.<sup>8</sup>

(20) Non-A-ISL version of long-distance Meeussen’s rule



Such a map is *not* A-ISL for the same reason that (17) is not ISL: on the tone tier the first H can be arbitrarily far away from the second, and so there is no finite QF statement that can identify the triggering environment. Thus the representational assumptions that affect computational classification go beyond the choice of string versus AR. We will see more cases like this in the analyses presented in §3.

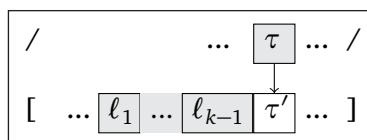
This example also highlights the limits to the expressive power of A-ISL. First, the map *on each tier* must be an ISL map. Notice in (18) that this process is essentially changing HH sequences to HL sequences, with everything else staying the same. The representation then isolates the otherwise long-distance process to an ISL function on the tone tier. Chandlee & Jardine formalize this in the following theorem.

(21) **Theorem.** If an AR map is A-ISL, then the individual map on each tier is ISL.  
 (Chandlee & Jardine 2019a: Thm 1)

A consequence of this theorem is that any modifications on the tonal tier are *independent* of any changes to the association relation. This means those processes for which a change in tone depends on information on both tiers are not A-ISL (though they may still be ISL, as we will see). In addition, given the way ISL and A-ISL are defined, they can only refer to *input* information. The results of Chandlee & Jardine (2019a), however, reveal cases of tone processes where this limitation is problematic, and for which a comparable notion of output locality is needed.

## 2.2 Output locality

In this section we propose two additional function classes that can reference output structure, one for strings and one for ARs. The function class for strings is based on the *output strictly local* (OSL) functions defined by Chandlee (2014) and Chandlee et al. (2015b), which provide a similar notion of locality as ISL, but in terms of the output string. As schematized in [Figure 2](#), with OSL functions there is some  $k$  such that, for any position in the string, its output is based on the previous  $k-1$  elements in the output. (The  $k^{\text{th}}$  element is the current input position.)



**Figure 2** Computing an OSL function. Here,  $\tau$  is an arbitrary input element;  $\tau'$  is its output. The sequence  $\ell_1, \ell_2, \dots, \ell_{k-1}$  is the information available to the function when determining  $\tau'$ .

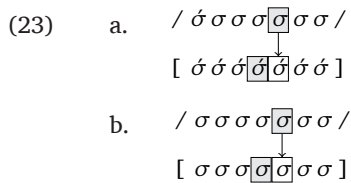
For example, consider a process that spreads a high tone to the end of the string, as in the rule in (22), applied iteratively.

(22)  $\sigma \rightarrow \acute{\sigma} / \acute{\sigma} \_$

<sup>8</sup> We thank an anonymous reviewer for pointing this out.



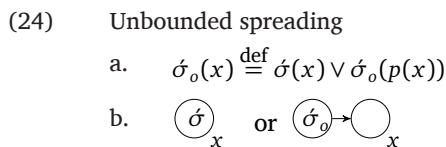
Two example mappings for this rule are shown in (23).



Examples (23a) and (23b) contrast two input  $\sigma$ 's (highlighted in boxes) one of which is output as  $\acute{\sigma}$  and one of which must remain  $\sigma$ . The surrounding environment of both  $\sigma$ 's in the input string is identical; it is the preceding item in the *output* string that differentiates them.<sup>9</sup> This is what makes the process OSL instead of ISL.

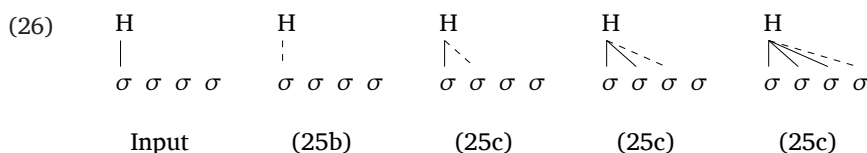
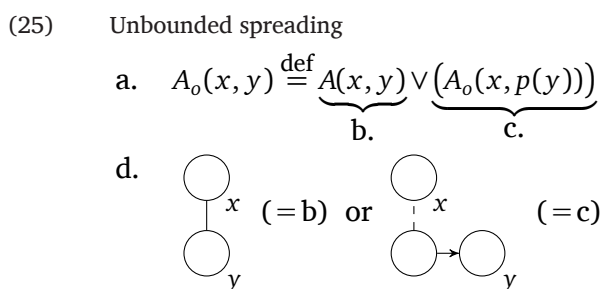
The OSL functions have been defined in terms of formal language theory and finite-state automata, but no logical characterization currently exists. We instead propose a logical characterization of two function classes we are calling *recursive strictly local (RSL)* and *autosegmental recursive strictly local (A-RSL)*, for reasons that will become apparent in a moment.

Following Koser et al. (2019) and Chandlee & Jardine (2019b), our definitions of RSL and A-RSL invoke the notion of *least-fixed point (LFP)* logics, which allow for recursive definitions of the logical formulas that make up the transductions. The full notation of LFP is technical, so we use *implicit definitions* (cf. Rogers 1998), that is, definitions that are explicitly recursive. For example, the process represented in (22) can be achieved with the formula in (24a) that identifies which output positions receive a high tone. (In the diagram we use  $\acute{\sigma}_o$  to denote the property of satisfying  $\acute{\sigma}_o(x)$ .)



Notice that this formula is recursive, in that it references itself in the second disjunct. The first disjunct then starts the recursion with any underlying high tone, which will remain high in the output. The second disjunct then checks the *output* label of a given position's predecessor, likewise taking on a high tone if the predecessor bears one in the output.

Over ARs instead of strings, unbounded spreading can be achieved using recursion in the output association relation, as in (25a). An example map is given in (26).



In (26), the H and the initial syllable are associated in the input; thus, they satisfy the first disjunct (25b) and remain associated in the output. Because the initial syllable is the predecessor of the second syllable, the H and the *second* syllable satisfy the recursive disjunct (25c), and

<sup>9</sup> Obviously if we look far enough back in the input string we will identify a  $\acute{\sigma}$  in example (23a) but not in (23b). But of course the unbounded nature of the process means we can never be certain of how far back we will have to look, and so we can't prespecify the size of the shaded region.

so the second syllable is also associated to H in the output. And likewise for the remaining syllables. In this way, the recursive nature of the definition captures the iterative addition of association lines.

This use of recursion is the crucial difference between RSL and ISL, since the latter cannot use recursive formulas. Recursion is, however, a powerful mechanism, and so its use must be limited in order to maintain the restrictiveness that locality provides.<sup>10</sup> However, recursive logical definitions—when limited to a QF FO language—can be restricted to a level of expressivity that is appropriate for phonological processes (Bhaskar et al. 2020). Our proposed definition of RSL and A-RSL then are QF FO logical transductions over strings and ARs, respectively, that satisfy the following conditions:

- (27)
- a. The transduction includes either formulas that use only *p* or formulas that use only *s* (i.e., *p* and *s* cannot be combined either in a single formula or across the set of formulas that make up the transduction).
  - b. Whenever a formula uses *p* or *s* it does so in a recursive predicate.

These conditions reflect the formal properties of the OSL functions in order to guarantee a similar degree of computational restrictiveness. We conjecture that the RSL functions will turn out to be the logical characterization of the OSL functions, though we leave a proof for future work.<sup>11</sup>

The reason for condition (27a) is that output locality (as defined by OSL functions) is necessarily *directional*. Given a formalism such as finite-state machines, for example, a different output string could be produced depending on whether the function reads the input string from *left-to-right* or *right-to-left*.<sup>12</sup> In our logical characterization, transductions that use only *p* simulate reading a string from left-to-right, and those that use only *s* simulate reading a string from right-to-left. It should be noted that the restriction in (27a) also excludes long-distance bidirectional processes, such as those discussed in Jardine (2016a). We bring this up again in §4.

The reason for condition (27b) is to limit the formula to only reference the input structure for the current position (i.e., *x* itself). Any other information it makes use of must be found in the output structure instead, as schematized in [Figure 2](#).

### 2.3 Autosegmental locality

Using these notions of input and output locality for string and AR maps, in the remainder of the paper we will survey a set of tone processes. We will classify these processes as ISL, A-ISL, RSL, and/or A-RSL, according to (28).

- (28)
- a. A process is **ISL** if it can be modeled as a non-recursive QF FO logical transduction over string representations.
  - b. A process is **A-ISL** if it can be modeled as a non-recursive QF FO logical transduction over ARs.
  - c. A process is **RSL** if it can be modeled as a QF FO logical transduction that satisfies the conditions in (27) over string representations.
  - d. A process is **A-RSL** if it can be modeled as a QF FO logical transduction that satisfies the conditions in (27) over ARs.

The results of the survey will demonstrate the extent to which these notions of locality overlap and the extent to which they distinguish different tone processes in terms of the computations they require. Furthermore they will reveal the various mechanisms by which ARs render non-local processes local, as well as the factors that can prevent them from doing so.

<sup>10</sup> For a more comprehensive explanation of the desirability of computational restrictiveness in phonology readers are referred to Chandlee & Heinz (2018).

<sup>11</sup> There is a third condition, which is not relevant to any of the examples analyzed in this paper and is technically involved, and so we omit it in the interest of clarity and for space limitations. Essentially though it serves to prohibit long-distance agreement, which can be enacted using least-fixed point logic via a type of covert 'markup'. See Chandlee & Jardine (2019b) for an example. As long-distance agreement is beyond the reach of the OSL functions, this ability of least-fixed point logic must be curbed in the logical characterization of OSL.

<sup>12</sup> This is not the case with ISL functions; when referencing only input the same output string will result regardless of direction. Hence the logical characterization of ISL is free to combine *p* and *s* in its formulas. Readers are referred to Chandlee (2014) for more on this difference between input and output locality.

### 3 Survey of tone processes

In this section we present analyses of a range of tone processes using the logical notions of locality defined in the previous section. For each process, we first analyze it assuming a string representation, and then again assuming an AR. Our string representations will be strings of TBUs, by default strings of syllables (e.g.,  $\sigma\sigma\sigma$ ) rather than strings of segments. One might question whether this representational choice hides some of the process’s computational complexity in the prior omitted step of parsing. But again, our goal is to isolate the process in order to assess its relative complexity in the domain under investigation (i.e., tone), and it is common practice to assume tone processes operate over TBUs instead of segments. In addition, given that the number of consonants that will appear between syllable nuclei is in fact bounded by the language’s phonotactic constraints, the complexity of parsing is not in fact believed to be that high.<sup>13</sup> Nonetheless, we acknowledge that an investigation into the computational complexity of various mechanisms for syllable parsing would not only be useful but may have implications for the results presented in this paper.

In addition, to simplify the diagrams, in the representations to follow (both strings and ARs) we will only include the word boundary symbols (#) when they are directly relevant to the pattern in question. This means that we largely abstract away from the question of which prosodic boundaries delineate the domain of the process. As a reviewer points out, this is an important consideration in tone, in particular as the often phrasal nature of tone poses a challenge for many theories of phonology (Sande et al. 2020). This paper does not attempt to wade into this complex problem, in the service of focusing on the basic distinction between local and non-local processes. It should be said, however, that model theory and logic can provide a flexible way of representing morphological and prosodic information and processes; interested readers are referred to Dolatian (2020).

Lastly, we would like to emphasize that in those cases where a process is amenable to more than one analysis (e.g., input and/or output local over strings and/or ARs), we will not be making an argument for which of those options is the best characterization. Identifying the ‘best’ analysis of any of the patterns presented here is not the goal. Rather it is to establish which computational properties the process does and does not have, in order to address the larger question of how representation interacts with these computational notions of locality.

The set of processes we analyze and their classifications are previewed in [Table 1](#). We begin in §3.1 with bounded and unbounded tone spread and shift.

PROCESS	ISL	A-ISL	RSL	A-RSL
Bounded spread (Bemba)	✓	✓	✗	✗
Bounded shift (Kuki-Thaadow)	✓	✓	✗	✗
Unbounded shift to penult (Zigula)	✗	✓	✗	✓
Unbounded spread to penult (Shambaa)	✗	✗	✗	✗
Unbounded Meeussen’s, deletion (Arusa)	✗	✓	✗	✗
Bounded Meeussen’s, lowering (Luganda)	✓	✗	✗	✗
Alternating Meeussen’s, lowering (Shona)	✗	✗	✓	✓

**Table 1** Summary of analyses.

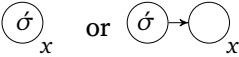
#### 3.1 Spread and shift

In bounded spread, an underlying tone spreads some fixed number of TBUs and no further. In Northern Bemba an H tone spreads exactly one TBU to the right; this is also referred to as ‘binary tone spread’ or ‘tone doubling’ (Bickmore & Kula 2013).

<sup>13</sup> Readers are referred to Strother-Garcia (2019) for more on the computational complexity of syllable representations and parsing.

- (29) *Northern Bemba* (Niger-Congo; Bickmore & Kula 2013)
- a. /tu-la-kak-a/      [tu-la-kak-a]      ‘we tie up’
  - b. /bá-la-kak-a/      [bá-lá-kak-a]      ‘they tie up’
  - c. /bá-ka-fik-a/      [bá-ká-fik-a]      ‘they will arrive’
  - d. /bá-ka-bil-a/      [bá-ká-bil-a]      ‘they will sew’

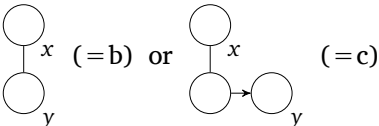
This process is intuitively local, and indeed it is ISL. A QF formula is given in (30a), with an example mapping given in (31).

- (30) Bounded spread (ISL)
- a.  $\acute{\sigma}_o(x) \stackrel{\text{def}}{=} \acute{\sigma}(x) \vee \acute{\sigma}(p(x))$
  - b. 

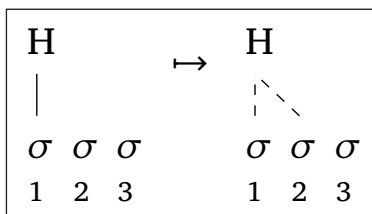
- (31)  $\acute{\sigma}\acute{\sigma}\acute{\sigma}\acute{\sigma} \mapsto \acute{\sigma}\acute{\sigma}\acute{\sigma}\acute{\sigma}$

The first disjunct of (30a) ( $\acute{\sigma}(x)$ ) identifies a syllable that is high in the input, such as the first syllable of (31). The second disjunct ( $\acute{\sigma}(p(x))$ ) identifies a syllable whose predecessor in the input is high, such as the second syllable. Both of these syllables then have a high tone in the output, achieving the effect of binary tone spread. The formula is QF and non-recursive; thus, bounded spread is ISL.

With an AR, the spreading rule corresponds to the definition in (32a), which reads as follows:  $x$  and  $y$  are associated in the output if and only if (32b)  $x$  and  $y$  are associated in the input; or (32c)  $x$  and the predecessor of  $y$  are associated in the input.<sup>14</sup> How this formula describes the mapping is illustrated with an example in [Figure 3](#).

- (32) a.  $A_o(x, y) \stackrel{\text{def}}{=} \underbrace{A(x, y)}_b \vee \underbrace{A(x, p(y))}_c$
- d. 

In [Figure 3](#), syllable 1 is associated to the H in the input, and so these positions satisfy the first disjunct  $A(x,y)$  and are also associated in the output. Additionally, as the H is associated to the predecessor of syllable 2 (namely, syllable 1), the H and syllable 2 satisfy the second disjunct  $A(x,p(y))$  and so are also associated in the output. As the H and syllable 3 satisfy neither disjunct, they are not associated in the output.



**Figure 3** Bounded spread formula in (32a) applied to an AR with three syllables.

Like (30a), the formula in (32a) is both QF and non-recursive. Bounded spreading is thus both ISL and A-ISL. Note that this analysis assumes underlying TBUs are either high-toned or unspecified. But the classification as ISL and A-ISL does not depend on this assumption. If for example the language instead had a H/L contrast, we would simply include a formula that

<sup>14</sup> Ternary spread (as in Copperbelt Bemba) is also ISL and A-ISL, as witnessed by the following formulas:

- (i)  $\acute{\sigma}_o(x) \stackrel{\text{def}}{=} \acute{\sigma}(x) \vee \acute{\sigma}(p(x)) \vee \acute{\sigma}(p(p(x)))$
- (ii)  $A_o(x, y) \stackrel{\text{def}}{=} A(x, y) \vee A(x, p(y)) \vee A(x, p(p(y)))$

These are simply the formulas in (30a) and (32a) with additional disjuncts for the syllable two syllables away. This analysis of ternary spread demonstrates how our notion of locality is not the same as the prior notion of ‘don’t count past two’. What makes a pattern local is not that it involves a window of size 2, it’s that an upper bound exists on the length of that window. For Copperbelt Bemba that length is 3. Formally, we’d say that Copperbelt Bemba’s pattern is a 3-ISL or 3-A-ISL function, while Northern Bemba’s is 2-ISL or 2-A-ISL.

specifies output TBUs as L-toned if they are L-toned in the input and their predecessors are *not* H-toned (similar to the analysis of Rimi in §2.1). Likewise in the AR analysis, the output association formula in (32a) would be expanded to make sure only high tones are spread. So while the analysis itself might look different depending on what tones are assumed to be present underlyingly, the *result* of the analysis is the same: ISL and A-ISL.

In the remainder of the section we will continue to follow the representational assumptions of the original cited analyses in those cases where the classification does not depend on those assumptions. In those cases where the classification is representation-dependent, however, we will make note of that fact.

As for output locality, bounded spread is in fact necessarily ISL, not RSL. This is because it requires keeping track of how far the spreading has gone, which in turn requires examining the input. Once the formula references the output structure the spreading will be necessarily unbounded, as the underlying trigger can no longer be distinguished from its targets. For similar reasons the process is also not A-RSL.

We now turn to bounded shift. An example is found in Kuki-Thaadow, in which a string of tones each associate to the following syllable, as in (33). The first and last tones also remain associated to their underlying TBUs.

- (33) *Kuki-Thaadow* (Tibeto-Burman; Hyman 2011)  
 a. /kà zóoŋ lien thúm/ [kà zòoŋ lien thũm] ‘my three big monkeys’  
 b. #ǝǝǝǝ# ↦ #ǝǝǝǝ#

As a map over strings, we need four formulas, one for each possible output tone. Output positions that bear a low tone satisfy the formula in (34a), which identifies two scenarios. One, the TBU is the first TBU and is underlyingly low (i.e.,  $\delta(x) \wedge \#(p(x))$ ). Two, the TBU is not the last TBU and its input predecessor is low (i.e.,  $\delta(p(x)) \wedge \neg\#(s(x))$ ). This captures the shift, excluding final TBUs which instead bear a contour tone.

- (34) Bounded shift in Kuki-Thaadow (ISL) (part 1)  
 a.  $\delta_o(x) \stackrel{\text{def}}{=} (\delta(x) \wedge \#(p(x))) \vee (\delta(p(x)) \wedge \neg\#(s(x)))$   
 b.

The parallel formula in (35a) designates those TBUs that bear a high tone in the output.

- (35) Bounded shift in Kuki-Thaadow (ISL) (part 2)  
 a.  $\acute{\sigma}_o(x) \stackrel{\text{def}}{=} (\acute{\sigma}(x) \wedge \#(p(x))) \vee (\acute{\sigma}(p(x)) \wedge \neg\#(s(x)))$   
 b.

The contour tones are handled by (36a) and (36c). (36a) says that a TBU bears a LH contour if it is the last TBU ( $\#(s(x))$ ), has a high tone in the input ( $\acute{\sigma}(x)$ ), and its input predecessor has a low tone ( $\delta(p(x))$ ). Similarly, (36c) designates output TBUs that bear a HL contour.

- (36) Bounded shift in Kuki-Thaadow (ISL) (part 3)  
 a.  $\delta_o(x) \stackrel{\text{def}}{=} \#(s(x)) \wedge \acute{\sigma}(x) \wedge \delta(p(x))$   
 b.
- c.  $\acute{\sigma}_o(x) \stackrel{\text{def}}{=} \#(s(x)) \wedge \delta(x) \wedge \acute{\sigma}(p(x))$   
 d.

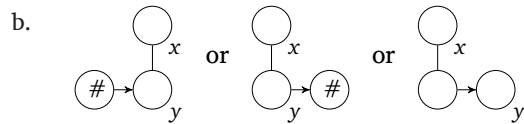
Collectively these formulas establish the map as ISL. It is also A-ISL. An AR map for the example in (33) is shown in (37), and the formula defining it is given in (38a).

- (37) 

#	L	H	L	H	#	↦	#	L	H	L	H	#
								└─┘	└─┘	└─┘	└─┘	
#	σ	σ	σ	σ	#		#	σ	σ	σ	σ	#

(38) Bounded shift in Kuki-Thaadow (A-ISL)

a.  $A_o(x, y) \stackrel{\text{def}}{=} ((\#(p(y)) \vee \#(s(y))) \wedge A(x, y)) \vee A(x, p(y)))$



Informally, (38a) says that a tone *x* and a TBU *y* should be associated in the output if at least one of the following holds: 1) they are associated in the input and *y* is either the first or last TBU, or 2) *x* is associated to *y*'s predecessor in the input.

Bounded shift then satisfies both of our notions of input locality. It does not, however, satisfy the corresponding notions of output locality. The condition on RSL/A-RSL formulas in (27b) forces them to look back to the output predecessor, but they need to see the input predecessor to determine which tone it bears. In addition, the formulas in (36) and (38a) violate condition (27a), as they reference both *s* and *p*. This is necessary for the specific version of shift found in Kuki-Thaadow in order to preserve the underlying tones on the first and last TBUs. Even without this added factor, though, the pattern is still not A-RSL for the reasons explained above.

We now turn to unbounded tone shift and spread. We already saw a case of unbounded shift in §1.1: in Zigula (Kenstowicz & Kisseberth 1990) an H shifts to the penultimate TBU. The examples in (4) are repeated in (39).

(39) *Zigula* (Niger-Congo; Kenstowicz & Kisseberth 1990)

- a. /ku-songoloz-a/ [ku-songoloz-a] 'to avoid'
- b. /á-songoloz-a/ [a-songolóz-a] 'he/she is avoiding'
- c. /ku-lómbez-a/ [ku-lombéz-a] 'to ask'
- d. /ku-lómbez-ez-a/ [ku-lombez-éz-a] 'to ask for'
- e. /ku-lómbez-ez-an-a/ [ku-lombez-ez-án-a] 'to ask for each other'

Not surprisingly, this map is not ISL. To see why, consider the contrast between the penultimate syllable in an underlyingly toneless word (40a) and toned words (40b).

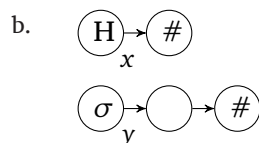
- (40) a.  $\sigma\sigma\sigma\sigma \mapsto \sigma\sigma\sigma\sigma$   
 b.  $\sigma\acute{\sigma}\sigma\sigma \mapsto \sigma\sigma\acute{\sigma}\sigma$   
 $\acute{\sigma}\sigma\sigma\sigma \mapsto \sigma\sigma\acute{\sigma}\sigma$

To correctly place the high tone in the output we need a formula that is false for the penult in (40a) but true for the penults in (40b). But without a quantifier, this formula must rely on a finite sequence  $\acute{\sigma}(p(x)) \vee \acute{\sigma}(p(p(x))) \vee \acute{\sigma}(p(p(p(x)))) \vee \dots$  that covers all possible distances from *x*, and this is not possible with unbounded phenomena.

With ARs, unbounded shift has been derived multiple ways, often by decomposing it into two steps such as spreading and delinking (see, for example, Kenstowicz & Kisseberth 1990; Odden 2001). Our analysis treats unbounded shift as a single function that directly associates the final H to the penultimate syllable.<sup>15</sup> The output association function defined in (6) and repeated in (41a) associates *x* and *y* in the output if and only if *x* is the final tone and *y* is the penultimate syllable. This shifts the association of the final H will to the penult, regardless of where it was associated in the input. An example is given in (42).

(41) Unbounded shift (A-ISL)

a.  $A_o(x, y) \stackrel{\text{def}}{=} H(x) \wedge \#(s(x)) \wedge \sigma(y) \wedge \#(s(s(y)))$



<sup>15</sup> Representing unbounded shift as a one-step process is reminiscent of previous OT analyses (e.g., Bickmore 1996; Yip 2002), in which an ALIGN constraint pulls the H tone to the right.





by a function that removes a word-final H (which is both (A-)ISL and (A-)RSL). We will say more about composing processes in §4.

We reiterate here that our goal is not to argue for a particular analysis, nor for a particular factoring of generalizations. Rather it is to assess which of our notions of locality each process satisfies. The case of Shambaa does, however, exemplify the way in which the computational framework advocated for here allows us to identify the computational distinctions among competing analyses and sets of assumptions.

### 3.2 Variants of Meeussen’s rule

Next we consider variants of Meeussen’s rule. First, in Arusa, the last H in a phrase is deleted following another H, no matter the distance (Levergood 1987; Odden 1994).

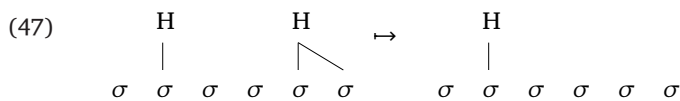
- (45) *Arusa* (Eastern Nilotic; Levergood 1987: 58)
- a. /sídáy/                    [sídáy]                    ‘good’
  - b. /enkér sídáy/            [enkér siday]            ‘good chair’
  - c. /olórika sídáy/           [olórika siday]           ‘good ewe’

In (45a), the underlying H’s in /sídáy/ ‘good’ surface faithfully, whereas in (45b) and (45c) they are deleted because of the presence of a H in the preceding word.<sup>16</sup> This deletion applies at any distance: in (45c) the triggering H is two TBUs away from the target.

Not surprisingly, this process is not ISL, for reasons that at this point should be clear. Consider the schematic examples in (46). The predicate  $\sigma_0(x)$  cannot detect the presence of the trigger H in (46b) and (46c) without a quantifier.

- (46)
- a.  $\sigma\sigma\sigma \mapsto \sigma\sigma\sigma$
  - b.  $\sigma\sigma\sigma \mapsto \sigma\sigma\sigma$
  - c.  $\sigma\sigma\sigma \mapsto \sigma\sigma\sigma$

However, it is A-ISL. An example mapping is shown in (47). With logical definitions, ‘deletion’ results when an input element does not satisfy *any* output formula. To capture deletion in Arusa, then, we write a formula that is *not* satisfied by H’s that are both final and preceded by another H. This is given in (48a).<sup>17</sup>



(48) Unbounded Meeussen’s Rule deletion in Arusa (A-ISL)

- a.  $H_0(x) \stackrel{\text{def}}{=} H(x) \wedge \neg(H(p(x)) \wedge \#(s(x)))$
- b.  $\textcircled{H}_x$  and not  $\textcircled{H} \rightarrow \textcircled{\phantom{H}} \rightarrow \textcircled{\#}$

This formula is interpreted as follows. The output label of  $x$  is H if and only if it satisfies two conditions. First,  $x$  must be an H in the input ( $H(x)$ ). Second, if  $x$  is the final member of its tier, it cannot be preceded by another H ( $\neg(H(p(x)) \wedge \#(s(x)))$ ).

Just as with the hypothetical long-distance Meeussen’s lowering rule presented in §2.1, however, this classification is dependent on an underlying H/ $\emptyset$  contrast. If the intervening TBUs between the two H’s are specified, it will no longer be possible to detect the presence of a triggering H using a finite number of calls to predecessor.

As for output locality, this process is not RSL for the same reason it is not ISL: the necessary information (preceding H tone) is not a bounded distance away regardless of whether you look at the input or output. And the formula in (48a) violates condition (27a) in that it needs to look

<sup>16</sup> The fact that both H’s are deleted from /sídáy/ is explained in an AR with a single H associated to two TBUs. The generalization would be complicated then in a string-based representation, in which each TBU must bear its own H. We put aside this complication, however, since even without it the rule is not ISL.

<sup>17</sup> Note the process only applies at the end of a phrase (Odden 1994), but for convenience we use # as a phrase boundary.

both backwards (to identify a preceding H tone) and forwards (to determine if x is phrase-final). So the process is also not A-RSL.

Next are two bounded variants of Meeussen’s Rule. In Luganda (Hyman & Katamba 2010), an underlying H lowers to L immediately following another H. Examples are shown in (49); the output forms are intermediate, before the application of other processes (see Hyman & Katamba 2010).

- (49) *Luganda* (Niger-Congo; Hyman & Katamba 2010)
- |    |                         |                       |                                |
|----|-------------------------|-----------------------|--------------------------------|
| a. | /a-láb-a/               | a-láb-a               | ‘s/he sees’                    |
| b. | /bá-láb-a/              | bá-làb-a              | ‘they see’                     |
| c. | /bá-lí-láb-a/           | bá-li-làb-a           | ‘they will see’                |
| d. | /a-bá-tá-lí-láb-il-ila/ | a-bá-tà-li-làb-il-ila | ‘they who will not look after’ |
| e. | /bá-ki-láb-a/           | bá-ki-láb-a           | ‘they see it’                  |

This process is ISL. Schematic examples are given in (50). Note in particular example (50c), in which two  $\acute{\sigma}$ s are lowered. This shows the input-oriented nature of the process, as only in the input does that second  $\acute{\sigma}$  follow another  $\acute{\sigma}$ .

- (50)
- |    |   |
|----|---|
| a. | $\sigma\acute{\sigma}\sigma \mapsto \sigma\acute{\sigma}\sigma$   |
| b. | $\acute{\sigma}\acute{\sigma}\sigma \mapsto \acute{\sigma}\grave{\sigma}\sigma$                             |
| c. | $\acute{\sigma}\acute{\sigma}\acute{\sigma}\sigma \mapsto \acute{\sigma}\grave{\sigma}\grave{\sigma}\sigma$ |
| d. | $\acute{\sigma}\acute{\sigma}\acute{\sigma}\sigma \mapsto \acute{\sigma}\acute{\sigma}\acute{\sigma}\sigma$ |

The logical transduction includes the predicates  $\acute{\sigma}_o(x)$  and  $\grave{\sigma}_o(x)$ , which specify when a syllable is realized as H-toned or L-toned, respectively.

- (51) Bounded Meeussen’s Rule in Luganda (ISL)

- |    |   |
|----|---|
| a. | $\acute{\sigma}_o(x) \stackrel{\text{def}}{=} \acute{\sigma}(x) \wedge \neg\acute{\sigma}(p(x))$                      |
| b. | $\textcircled{\neg\acute{\sigma}} \rightarrow \textcircled{\acute{\sigma}}_x$   |
| c. | $\grave{\sigma}_o(x) \stackrel{\text{def}}{=} \grave{\sigma}(x) \vee (\acute{\sigma}(x) \wedge \acute{\sigma}(p(x)))$ |
| d. | $\textcircled{\acute{\sigma}} \rightarrow \textcircled{\acute{\sigma}}_x$   |

(51a) specifies that a TBU is output as H-toned iff it is H-toned in the input ( $\acute{\sigma}(x)$ ) and its predecessor is not H-toned in the input ( $\neg\acute{\sigma}(p(x))$ ). An example with indexed TBUs is given below. In (52) TBUs 1 and 4 satisfy  $\acute{\sigma}_o(x)$ , as they satisfy both  $\acute{\sigma}(x)$  and  $\neg\acute{\sigma}(p(x))$ —1 because it is the first TBU, and 4 because its predecessor is  $\sigma$ . Thus, 1’ and 4’ are labeled  $\acute{\sigma}$  in the output.

- (52)
- |                  |                  |          |                  |          |                  |
|------------------|------------------|----------|------------------|----------|------------------|
| $\acute{\sigma}$ | $\acute{\sigma}$ | $\sigma$ | $\acute{\sigma}$ | $\sigma$ | $\acute{\sigma}$ |
| 1                | 2                | 3        | 4                | 1’       | 2’               |
|                  |                  |          |                  | 3’       | 4’               |

In contrast, TBU 2 instead satisfies (51c)  $\grave{\sigma}_o(x)$ , which specifies x is H-toned in the input and its predecessor is also H-toned ( $\acute{\sigma}(p(x))$ ).

Over ARs, the desired map is as exemplified in (53).

- (53)
- |    |          |          |           |           |          |          |
|----|----------|----------|-----------|-----------|----------|----------|
| a. | H        | H        | $\mapsto$ | H         | L        |          |
|    |          |          |           |           |          |          |
|    | $\sigma$ | $\sigma$ |           | $\sigma$  | $\sigma$ |          |
| b. | H        |          | H         | $\mapsto$ | H        | H        |
|    |          |          |           |           |          |          |
|    | $\sigma$ | $\sigma$ | $\sigma$  |           | $\sigma$ | $\sigma$ |

Both the second H in (53a) and the second H in (53b) follow an H in the input, but the lowering should only apply to (53a). Here we recall the theorem in (21), which asserts that for a process to be A-ISL, any changes on a particular tier must *only* use information that is local in the input *on that tier*. The Luganda map is not local in this way; whether or not an H changes to L depends both on whether or not there is a preceding H *and* whether their associated TBUs are adjacent on the TBU tier. But this requires a quantifier, as shown in (54a).

- (54) a.  $L_o(x) \stackrel{\text{def}}{=} H(x) \wedge H(p(x)) \wedge (\exists y)[A(x, y) \wedge A(p(x), p(y))]$
- b.

This is then our first case of a map that is ISL but not A-ISL, or in other words, there are local maps that ARs actually render *non-local*.

Again we note here that this analysis is representation-dependent. For example, if the intervening unspecified TBUs receive some sort of representation on the tonal tier, the map would become A-ISL.<sup>18</sup>

- (55) a.  $\begin{array}{ccc} H & H & \\ | & | & \\ \sigma & \sigma & \sigma \end{array} \mapsto \begin{array}{ccc} H & L & \\ | & | & \\ \sigma & \sigma & \sigma \end{array}$
- b.  $\begin{array}{ccc} H & \emptyset & H \\ | & & | \\ \sigma & \sigma & \sigma \end{array} \mapsto \begin{array}{ccc} H & & H \\ | & & | \\ \sigma & \sigma & \sigma \end{array}$

In (55b), an explicit unspecified symbol  $\emptyset$  marks the intervening unspecified TBUs. Given this, the two examples can easily be distinguished again by simply examining the predecessor of the second H. Explicit unspecified marks on the tonal tier are not standard representation, and so we do not propose this as a solution to the problem. This example does, however, offer an interesting case study into the interaction of locality and representation. Given locality as defined presently, the Luganda process is local over strings regardless of underlying tone specifications, but it is only local over ARs given a particular representation of underlying tones. We will return to this point in the summary at the end of this section.

It was noted above that this process is necessarily input-oriented, in that the conditions for lowering need to be checked in the input form. Not surprisingly then, the process isn't RSL or A-RSL. Consider again example (50c). In a RSL analysis, the formula for  $\delta_o(x)$  would be forced by condition (27b) to look at the output value of the predecessor. For the third TBU in (50c) the output value of its predecessor is  $\delta$ , and so the conditions for lowering are not met. Furthermore, the map is also not A-RSL for the same reason it wasn't A-ISL: determining whether two tones are associated to adjacent TBUs requires a quantifier regardless of whether you look at the input or the output.

Lastly, we consider another variant of Meeussen's that produces an alternating pattern of H and L tones. In Shona, an H is lowered following another H. In contrast to Luganda, however, an H tone to which Meeussen's Rule has applied does not serve as a trigger for a following H tone. Examples are given in (56).

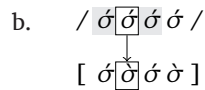
- (56) *Shona* (Niger-Congo; Odden 1986)
- |    |               |               |                     |
|----|---------------|---------------|---------------------|
| a. | /hóvé/        | [hóvé]        | 'fish'              |
| b. | /né-hóvé/     | [né-hòvè]     | 'with-fish'         |
| c. | /né-é-hóvé/   | [né-è-hóvé]   | 'with-of-fish'      |
| d. | /né-é-é-hóvé/ | [né-è-é-hòvè] | 'like-with-of-fish' |

In (56), the underlying H tones in /hóvé/ 'fish' surface as L when following the H-toned prefix /né-/ 'with' in (56b). This lowering is blocked when the H-toned prefix /é-/ 'of' intervenes in (56c). Instead, the H tone of this prefix lowers.

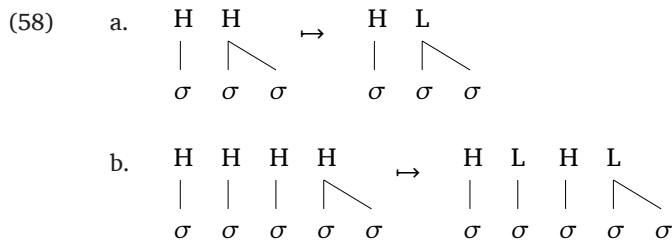
This map is not ISL. To see why, compare the outputs for the second and third H-toned syllables in the four syllable word in (57).

- (57) a.  $\begin{array}{c} / \acute{o} \acute{o} \acute{o} \acute{o} / \\ \downarrow \\ [ \acute{o} \grave{o} \acute{o} \grave{o} ] \end{array}$

18 We thank an anonymous reviewer for pointing this out.



The third syllable has an H tone in the input, and its predecessor and successor are also H-toned. But the exact same conditions hold for the second syllable, which instead is output as L-toned. Clearly, the input string does not contain the needed information to distinguish these two output positions. For the same reason, this pattern is not A-ISL. As shown in (58), the input string of the tonal tier does not sufficiently distinguish the H's that lower from those that do not.



Clearly, the crucial information is available provided we can access the output structure. As a map over strings then, alternating Meeussen's is RSL. The formulas in (59) assert which syllables bear high and low tones.

- (59) Alternating Meeussen's Rule in Shona (RSL)
- a.  $\acute{\sigma}_o(x) \stackrel{\text{def}}{=} \acute{\sigma}(x) \wedge (\#_o(p(x)) \vee \grave{\sigma}_o(p(x)))$
  - b.  $\textcircled{\grave{\sigma}}_o \rightarrow \textcircled{\acute{\sigma}}_x$
  - c.  $\grave{\sigma}_o(x) \stackrel{\text{def}}{=} \grave{\sigma}(x) \vee (\acute{\sigma}(x) \wedge \acute{\sigma}_o(p(x)))$
  - d.  $\textcircled{\acute{\sigma}}_x \text{ or } \textcircled{\acute{\sigma}}_o \rightarrow \textcircled{\acute{\sigma}}_x$

In both cases these formulas determine the output tone by looking at the *output* tone of the TBU's predecessor: if the predecessor is H-toned the TBU is L-toned, and vice versa. (Underlying L-tones remain L.) Consider a string of four H-toned TBUs,  $\acute{\sigma}\acute{\sigma}\acute{\sigma}\acute{\sigma}$ , which should be mapped to  $\acute{\sigma}\grave{\sigma}\acute{\sigma}\grave{\sigma}$ . The first and third TBUs remain high in the output because they satisfy  $\acute{\sigma}_o(x)$ : they are high in the input and their predecessor in the output is either low or they are the first TBU. In contrast, the second and fourth TBUs are low in the output because they do not satisfy  $\acute{\sigma}_o(x)$  but they do satisfy  $\grave{\sigma}_o(x)$ : they are high in the input *and* their predecessors in the output are also high.<sup>19</sup>

Likewise, the map is also A-RSL, as the changes that take place on the tone tier are essentially the same as in the string-based map. The formulas just need to refer to H and L instead of  $\acute{\sigma}$  and  $\grave{\sigma}$ , as in (60). This is because the changes only take place on the tonal tier and do not affect the associations of tones to TBUs.

- (60) Alternating Meeussen's Rule in Shona (A-RSL)
- a.  $\text{H}_o(x) \stackrel{\text{def}}{=} \text{H}(x) \wedge \text{L}_o(p(x))$
  - b.  $\textcircled{\text{L}}_o \rightarrow \textcircled{\text{H}}_x$
  - c.  $\text{L}_o(x) \stackrel{\text{def}}{=} \text{L}(x) \vee (\text{H}(x) \wedge \text{H}_o(p(x)))$
  - d.  $\textcircled{\text{L}}_x \text{ or } \textcircled{\text{H}}_o \rightarrow \textcircled{\text{H}}_x$

<sup>19</sup> In contrast, an input like  $/\acute{\sigma}\grave{\sigma}\acute{\sigma}\acute{\sigma}/$  would not change. This time the first and fourth TBUs satisfy  $\acute{\sigma}_o(x)$ , and the second and third TBUs satisfy  $\grave{\sigma}_o(x)$ . For the latter, just being low in the input is sufficient (by the first disjunct) to remain low in the output.

### 3.3 Interim summary

**Table 2** repeats the summary of the results of our analyses. What’s clear from these results is that 1) both input and output locality play a role in the typology of tone patterns, as neither subsumes the other. In addition, both strings and ARs appear to make distinct contributions to locality, as there exist maps that are only local with each representation and not the other. Furthermore, there are multiple reasons why a particular map does not belong to a particular class. Some maps (e.g., Bemba, Kuki-Thaadow, Luganda, and Shona) are necessarily ISL/A-ISL or RSL/A-RSL because they require access to the input or output, respectively. Some maps (e.g., Zigula and Arusa) are necessarily A-ISL or A-RSL because over strings (but not ARs) the trigger is an unbounded number of positions away. In other cases, a map violates the formal constraints on the function class, such as RSL/A-RSL’s prohibition against looking in both directions (e.g., Shambaa and Arusa) or A-ISL/A-RSL’s restriction on using cross-tier information (e.g., Luganda).

PROCESS	ISL	A-ISL	RSL	A-RSL
Bounded spread (Bemba)	✓	✓	✗	✗
Bounded shift (Kuki-Thaadow)	✓	✓	✗	✗
Unbounded shift to penult (Zigula)	✗	✓	✗	✓
Unbounded spread to penult (Shambaa)	✗	✗	✗	✗
Unbounded Meeussen’s, deletion (Arusa)	✗	✓	✗	✗
Bounded Meeussen’s, lowering (Luganda)	✓	✗	✗	✗
Alternating Meeussen’s, lowering (Shona)	✗	✗	✓	✓

**Table 2** Summary of analyses.

In addition, the survey identified three cases—Zigula, Arusa, and Luganda—in which the classification over ARs depended on another representational assumption, namely what and how tones are marked underlyingly. Such cases demonstrate how this computational framework allows us to clearly enumerate the options for preserving locality. For example, in Luganda, do we introduce an explicit unspecified symbol on the tone tier in order to preserve locality over ARs? Or do we dispense with ARs and instead use a string analysis? Option three of course is to dispense with locality altogether and recognize the process as a ‘truly’ non-local phenomenon. We will not provide an answer for Luganda itself here. Rather, its example demonstrates how the precision of this computational framework facilitates the demarcation of local versus non-local and clarifies how that line is affected by representational choices.

At the outset of the paper we noted a goal of narrowing down the necessary categories for a computational theory of tone. Doing so requires further investigation into the gaps in the table in terms of what classes overlap others. Specifically, what’s missing are examples of processes in the following categories. One, a process that is only RSL, as the one example of an RSL map is also A-RSL. This would be a process that requires reference to the output structure, but for which the AR disrupts that output-locality (as in Luganda, in which input-locality over ARs was not possible without cross-tier information). And two, a process that is only A-RSL, as every A-RSL map is also either A-ISL or RSL. This would be a process that again necessarily references the output structure, but for which the trigger is an unbounded number of positions away given a string representation.

These gaps may be filled in by further investigation, or they may be accidental. Further study will thus aim to expand on this catalog of tone processes. While the non-existence of a particular process cannot be proven, as the catalog becomes more exhaustive the continued absence of processes in these categories will build confidence in a proposal that they can be omitted from a computational theory of tone.

## 4 Discussion

Our survey of tone processes revealed that the traditional distinction between local and long-distance phenomena is more nuanced when assessed in terms of the computational notions of

input and output locality defined at the outset of the paper. The ways in which locality interacts with representation is not as straightforward as one might think. This section considers the broader implications of these results for phonological theory. In §4.1, we review the import that locality has had in the phonological literature and compare our formal definitions of locality with previous conceptions. Then in §4.2, we discuss how the results presented here can be developed into a complete computational theory of tone processes.

#### 4.1 Previous considerations of locality and representation

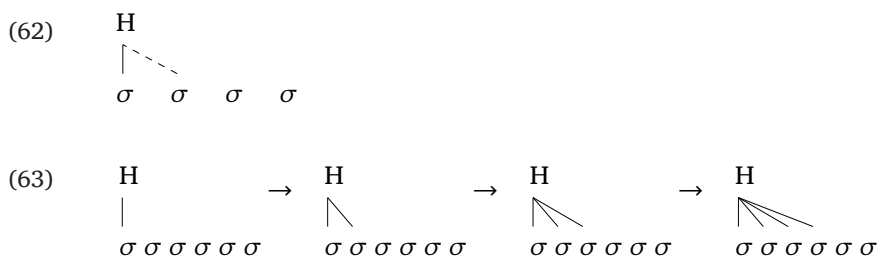
Considerations of locality have loomed large in the literature on phonological representations. Odden (1995: 474) states:

A widely held desideratum in phonological theory—indeed much of the motivation for nonlinear phonology and one of the outstanding problems of linear phonology—is that rules should be ‘local.’ Though there are many unresolved problems in the locality issue, it is generally agreed that a local rule formulation would only allow specification of one element to the right and/or left of the focus.

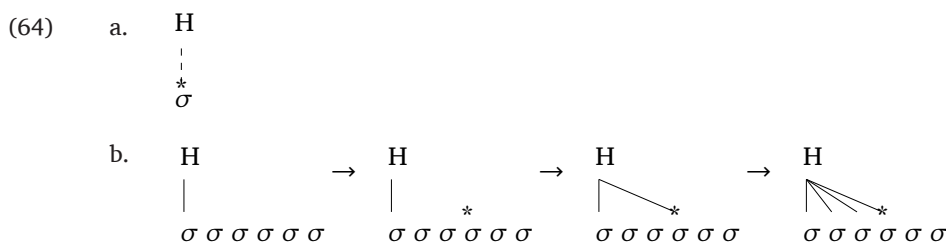
As an example, Odden raises the process of unbounded spread to antepenultimate position in the Nguni languages, such as Ndebele (Sibanda 2004; Hyman 2011):

- (61) *Ndebele* (Niger-Congo; Sibanda 2004; Hyman 2011)
- |    |                     |                     |                         |
|----|---------------------|---------------------|-------------------------|
| a. | /ú-ku-hlek-a/       | [ú-kú-hlek-a]       | ‘to laugh’              |
| b. | /ú-ku-hlek-is-a/    | [ú-kú-hlék-is-a]    | ‘to amuse (make laugh)’ |
| c. | /ú-ku-hlek-is-an-a/ | [ú-kú-hlék-ís-an-a] | ‘to amuse each other’   |

Odden notes that it is possible to posit the iterative rule in (62) for this process. The target for the rule is a syllable that is followed by two other syllables. As shown in (63), this will iterate up to but not including the penult, which is followed by only one syllable.



However, this rule is non-local according to Odden’s definition, because it refers to more than one syllable to the target’s right. Instead, Odden argues that an accentual analysis makes such rules local. If the antepenultimate syllable receives an accent, then we can formulate a tone-accent attraction rule, as in (64a), which associates the H and the antepenult. While Odden does not explicitly state how spreading would be accomplished, one could imagine that the Well-Formedness Condition (WFC; Goldsmith 1976) would apply and fill in the intervening TBUs, as in the derivation in (64b).



The rule in (64a) is local, according to Odden (1995)’s definition, as it refers only to the target. However, there is a bit of a trick here: it creates a gapped structure in between two non-local TBUs—the initial and antepenult syllables—over which the WFC then operates.

This concern for locality has been no less important for Optimality Theory (OT; Prince & Smolensky 2004). As McCarthy (2010: 200) states, one of the ‘fundamental descriptive and explanatory goals of OT’ is ‘to derive complex patterns from the interaction of simple constraints’.

One such concrete proposal is the articulatory notion of ‘strict locality,’ meaning articulatory locality at the level of the segment (Gafos 1996; Chioşain & Padgett 2001). Under such a theory, spreading is motivated by output markedness constraints grounded by the phonetics governing the articulation of adjacent segments.

However, not all tone patterns can be motivated entirely by local output conditions. A case in point is that of bounded shift in Rimi (Myers 1997), discussed in §2.1 and repeated below in (65).

(65) *Rimi* (Schadeberg 1979; Myers 1997)

a.	/u-hang-a/	[u-hang-a]	‘to meet’
b.	/u-pùm-a/	[u-pùm-á]	‘to go away’
c.	/mu-ntu/	[mu-ntu]	‘person’
d.	/rá-mu-ntu/	[ra-mú-ntu]	‘of a person’
e.	/u-huvi-í/	[u-huvi-í]	‘belief’
f.	/mu-tém-í/	[mu-tem-í]	‘chief’

Crucial to capturing this shift is the position of the underlying H tone. To handle this, Myers (1997) posits the following LOCALITY constraint.

(66) LOCAL (Myers 1997: 876, (50)): If an input tone  $T$  has an output correspondent  $T'$ , some edge of  $T$  must correspond to some edge of  $T'$ .

Importantly, this constraint is not a markedness condition, as it crucially refers to the input. In fact, it is more of a ‘two-level’ constraint (McCarthy 1996; Kager 1999), in that it refers both to the input position in which a tone appears in the UR as well as the output position in which it surfaces in the SR. The need for such constraints in an analysis of Rimi in OT, which is classically output-oriented, again shows that the pattern is input-oriented. That nature of the pattern is confirmed with our analysis that it is ISL and A-ISL but not RSL/A-RSL. And its input orientation holds independently of the use of ARs versus strings. In this way we see that the nature of a given tone pattern has at least two facets: its locality (related to the need for ARs or not to render it local) and its orientation with respect to input- or output-based computation.

Furthermore, output constraints often invoked to explain tone patterns are not ‘local’ by the definitions already put forth. Much of the foundational work on tone in OT (Myers 1997; Cassimjee & Kisseberth 1998; Yip 2002) makes copious use of gradient ALIGN constraints (McCarthy & Prince 1993; 1995).<sup>20</sup> A gradient ALIGN constraint counts the distance between two elements in a representation, which is difficult to fit into a conception of ‘local’. In computational terms, this kind of constraint is in fact quite complex (Eisner 1997b). While proposals to replace ALIGN with local constraints exist (Eisner 1997a; McCarthy 2003), they have not been widely adopted. A logical perspective like the one put forth in this paper presents a possible solution to this, as restricted logics can be used as a local constraint definition language (Jardine & Heinz 2016). However, this is not a trivial issue, as optimization can generate non-local patterns purely from computationally local constraints (Gerdemann & Hulden 2012; Koser & Jardine 2020).

## 4.2 A computational theory of tone

The results of our survey increase our understanding of how ARs relate to locality in two key ways: 1) they highlight *which* types of processes are made local by ARs, namely exactly those for which the alternation depends on local information within, but not across, tiers, and 2) they show that both input and output locality are needed in a comprehensive computational theory of tone. Yet, while the A-ISL and A-RSL properties contribute to our understanding of locality in tone, several key questions remain.

First A-ISL and A-RSL do not cover all attested tone patterns. One example in our survey was local Meeussen’s rule in Luganda, which was shown in §3.2 not to be A-ISL because it required reference to both tiers at once. One interpretation of this result is that both ISL and A-ISL are necessary to capture tone. Indeed, what this makes clear is that local information on the TBU tier—specifically, the tone values of neighboring TBUs—is also necessary to capture at least some processes. Another way of interpreting this result is that autosegmental locality would be better defined using some representation other than our naïve implementation of ARs. As one

<sup>20</sup> Zoll (2003) shows how many tone patterns can be explained without ALIGN constraints, but does not completely do away with them.



reviewer points out, we could define some representation that includes both the autosegmental tonal tier and the local tonal information on the TBU tier. Indeed, Jardine (2020)'s recent work on locality in tonal phonotactics uses grammars that reference both in tandem. Thus, a unified computational theory of tone is likely possible given the right kind of representation.

Another example that escaped our definition of locality was the unbounded spreading pattern in Shambaa. We showed in §2.2 that 'generic' unbounded spreading is straightforwardly RSL and A-RSL. But unbounded spread in Shambaa is not, because it requires 'looking ahead' to determine whether it has reached the penultimate syllable (and again, (A-)RSL forbids any lookahead).

There are two possible solutions to this. One is to appeal to a function class that combines these two abilities, which is precisely what the *input-output strictly local (IOSL) functions* do (Chandlee et al., in prep.). Briefly, as their name suggests, the IOSL functions allow reference to local information both in the input and in the output. Space limitations prevent a thorough description and formalization of this class, but in future work they may provide a useful means of investigating the interaction of tone processes and prosodic or metrical structure.<sup>21</sup>

A second solution is to break the process down into the composition of an RSL function and an ISL function, as already discussed at the end of §3.1. Such a decomposition strategy may allow other processes to be brought into the A-ISL and A-RSL fold. As discussed in 2.2, the unidirectional nature of RSL and A-RSL exclude bidirectional processes such as those highlighted in Jardine (2016a). Briefly, Jardine (2016a) points out the computational complexity of *unbounded tone plateauing*, in which two H tones in a domain merge and associate to all TBUs in between them, no matter how far apart they are. While considerations of space preclude a detailed discussion, this process could be made computationally local by decomposing it into an A-ISL merging process composed with an A-RSL spreading process.

This is related to another big question, which is how a computational theory can combine individual processes to generate a full phonological grammar. In Appendix A we show how multiple processes can be combined directly into a single logical transduction, but this is only a start. An obvious option for combining functions from different classes is functional composition. However, composition is potentially very powerful, as computational properties may not be preserved under composition. Neither the ISL class nor the OSL class as originally defined is closed under composition, meaning that the composition of two ISL processes is not always itself ISL. Future work can and should study under what conditions A-ISL and A-RSL are preserved under composition and when they are not. Additionally, other operations for combining functions, such as priority union (Karttunen 1998) can and should be investigated. Finally, as noted in §3 a full grammar will also require reference to prosodic structure (Dolatian 2020). As noted there, it is possible to write a grammar that references this kind of representation, including the building of and translation between different kinds of structures.

Lastly, as with segmental phonology, we could say that patterns like Shambaa's unbounded spreading and unbounded tone plateauing are simply non-local and require a different mechanism for computing long-distance dependencies. This raises new questions in terms of whether and how to restrict those dependencies and what their existence predicts typologically. For example, a reviewer points out that 'long-distance' Meeussen's (as in (19)) is only A-ISL if there is an underlying H/∅ contrast, as any intervening tones marked on the tone tier would disrupt the locality of the two H's. If all tone patterns are local in the sense defined in this paper, this in turn predicts that such 'long-distance' effects are only possible in systems with privative tone. Whether such predictions turn out to be true is an interesting question we leave for future work.

## 5 Conclusion

This paper applied computational techniques to study, in a rigorous way, what exactly ARs contribute to phonological locality. The framework provided here allows us to formally

<sup>21</sup> See Chandlee (2019) for an application of the IOSL functions to an interaction of tone sandhi processes in Tianjin Chinese. See also Graf & Mayer (2018) for an approach to Sanskrit n-retroflexion that uses an IOSL function to project segments onto a tier over which well-formedness can then be assessed.

distinguish which tone processes are and are not local, and which are only made local with ARs (e.g., Meeussen's in Arusa is neither ISL nor RSL, but it is A-ISL) and those that are *not* local over ARs (such as unbounded spreading in Shambaa). In addition, it draws a clear distinction between input locality and output locality, both of which appear to be sufficient conditions to accommodate the full typology of tone. Lastly, the survey also revealed how the interaction of the tone process with other factors like metrical structure can also affect its locality. Consider again Shambaa, which is neither input nor output local by the definitions used in this paper. If, however, the representation including structural marking (e.g., penultimate), then the process can be analyzed as local. Such results may have implications for how the different components of a phonological grammar (i.e., maps, representations, structure) are learned, in particular how their varying and interacting contributions to locality may demand distinct learning mechanisms and/or factored approaches to learning (see e.g., Heinz & Idsardi 2013; Heinz & Rogers 2013).

The computational properties used in our analyses make precise what it means to be local over different representations and serve as the foundation for a computational theory of tone that requires more than one notion of locality. Such a theory will ultimately narrow in on the minimal number of computational categories needed to accommodate the full range of tone processes. What those categories are is a question we don't yet have a complete answer to, because this investigation has raised some non-trivial questions that require further study. In addition, we have established here a framework through which the computational properties of competing analyses grounded in non-computational theories can be assessed. If a particular process is local by our definitions under one set of assumptions but not another, that fact can be used—alongside several other considerations such as how that process fits into the larger grammar—to argue for or against a particular analysis.

Looking forward then, further study will aim to fit the complete typology of tone processes into the provided framework, investigating the full scope of what happens in tone. This includes those processes that have proven to be difficult for most if not all approaches, such as interactions between tone and segmental features (e.g., Odden 2011; Hyman 2014b). Fitting such processes into the framework provided here allows for their apparent exceptionality to be either confirmed or resolved, again identifying in a well-defined sense how they differ from other tone processes and how those differences relate to the interaction of locality and representation.

## Additional file

The additional file for this article can be found as follows:

- **Appendix A.** A unified analysis of OCP effects in Shona. DOI: <https://doi.org/10.5334/gjgl.1423.s1>

## Competing interests

The authors have no competing interests to declare.

## Author affiliations

**Jane Chandlee**  [orcid.org/0000-0001-9143-1305](https://orcid.org/0000-0001-9143-1305)  
Haverford College, US

**Adam Jardine**  [orcid.org/0000-0001-9436-8014](https://orcid.org/0000-0001-9436-8014)  
Rutgers University, US

## References

- Ao, Benjamin. 1991. Kikongo nasal harmony and context-sensitive underspecification. *Linguistic Inquiry* 22(2). 193–196.
- Berstel, Jean. 1982. Fonctions rationnelles et addition [Rational functions and addition]. In *Actes de l'école de printemps de théorie des langages [Proceedings of the spring school of language theory]*, 177–183. Laboratoire d'Informatique de Paris.

- Bhaskar, Siddharth, Jane Chandlee, Adam Jardine & Christopher Oakden. 2020. Boolean monadic recursive schemes as a logical characterization of the subsequential functions. In Alberto Leporati, Carlos Martín-Vide, Dana Shapira & Claudio Zandron (eds.), *Language and automata theory and applications – LATA 2020* (Lecture Notes in Computer Science), 157–169. New York: Springer. DOI: [https://doi.org/10.1007/978-3-030-40608-0\\_10](https://doi.org/10.1007/978-3-030-40608-0_10)
- Bickmore, Lee S. 1996. Bantu tone spreading and displacement as alignment and minimal misalignment. ROA #161-1196.
- Bickmore, Lee S. & Nancy C. Kula. 2013. Ternary spreading and the OCP in Copperbelt Bemba. *Studies in African Linguistics* 42. 101–132.
- Cassimjee, Farida & Charles Kisseberth. 1998. Optimal domains theory and Bantu tonology. In Charles Kisseberth & Larry Hyman (eds.), *Theoretical aspects of Bantu tone*, 265–314. CSLI.
- Chandlee, Jane. 2014. *Strictly local phonological processes*: University of Delaware dissertation.
- Chandlee, Jane. 2019. A computational account of tone sandhi interaction. In Katherine Hout, Anna Mai, Adam McCollum, Sharon Rose & Matthew Zaslansky (eds.), *Proceedings of the 2018 annual meeting on phonology*. Linguistic Society of America. DOI: <https://doi.org/10.3765/amp.v7i0.4572>
- Chandlee, Jane & Adam Jardine. 2019a. Autosegmental input strictly local functions. *Transactions of the Association for Computational Linguistics* 7. 157–168. DOI: [https://doi.org/10.1162/tac\\_l\\_a\\_00260](https://doi.org/10.1162/tac_l_a_00260)
- Chandlee, Jane & Adam Jardine. 2019b. Quantifier-free least fixed point functions for phonology. In *Proceedings of the 16th meeting on the mathematics of language (MOL 16)*, 50–62. Toronto, Canada: Association for Computational Linguistics. DOI: <https://doi.org/10.18653/v1/W19-5705>
- Chandlee, Jane, Adam Jardine & Jeffrey Heinz. 2015b. Learning repairs for marked structures. In Adam Albright & Michelle A. Fullwood (eds.), *Supplemental proceedings of the 2014 annual meeting on phonology*. Linguistic Society of America. DOI: <https://doi.org/10.3765/amp.v2i0.3760>
- Chandlee, Jane & Jeffrey Heinz. 2018. Strict locality and phonological maps. *Linguistic Inquiry* 49. 23–60. DOI: [https://doi.org/10.1162/LING\\_a\\_00265](https://doi.org/10.1162/LING_a_00265)
- Chandlee, Jane, Rémi Eyraud & Jeffrey Heinz. 2014. Learning strictly local subsequential functions. *Transactions of the Association for Computational Linguistics* 2. 491–503. DOI: [https://doi.org/10.1162/tac\\_l\\_a\\_00198](https://doi.org/10.1162/tac_l_a_00198)
- Chandlee, Jane, Rémi Eyraud & Jeffrey Heinz. 2015a. Output strictly local functions. In Andras Kornai & Marco Kuhlmann (eds.), *Proceedings of the 14th meeting on the mathematics of language (MOL 14)*, 52–63. Chicago, IL: Association for Computational Linguistics. DOI: <https://doi.org/10.3115/v1/W15-2310>
- Chandlee, Jane, Rémi Eyraud & Jeffrey Heinz. in prep. Input-output strictly local functions and their efficient learnability.
- Chiośain, Maire Ní & Jaye Padgett. 2001. Markedness, segment realization, and locality in spreading. In Linda Lombardi (ed.), *Segmental phonology in optimality theory*, 118–156. Cambridge: Cambridge University Press. DOI: <https://doi.org/10.1017/CBO9780511570582.005>
- Clements, George N. 1977. Neutral vowels in Hungarian vowel harmony: an autosegmental interpretation. In *Proceedings of NELS 7*, 49–64.
- Dolatian, Hossep. 2020. *Computing cyclic phonology in Armenian: Logical structure of the morphology-phonology interface*: Stony Brook University dissertation.
- Eisner, Jason. 1997a. Efficient generation in primitive Optimality Theory. In *Proceedings of the 35th annual meeting of the association for computational linguistics (ACL)*, 313–320. Madrid: Association for Computational Linguistics. DOI: <https://doi.org/10.3115/976909.979657>
- Eisner, Jason. 1997b. What constraints should OT allow? Talk handout, Linguistic Society of America, Chicago. ROA#204-0797. Available at <http://roa.rutgers.edu/>.
- Engelfriet, Joost & Hendrik Jan Hoogeboom. 2001. MSO definable string transductions and two-way finite-state transducers. *ACM Transactions on Computational Logic* 1. 1–38. DOI: <https://doi.org/10.1145/371316.371512>
- Gafos, Adamantios. 1996. *The articulatory basis of locality in phonology*: Johns Hopkins University dissertation.
- Gerdemann, Dale & Mans Hulden. 2012. Practical finite state Optimality Theory. In *Proceedings of the 10th international workshop on finite state methods and natural language processing*, 10–19. Donostia–San Sebastián: Association for Computational Linguistics.
- Goldsmith, John. 1976. *Autosegmental phonology*: Massachusetts Institute of Technology dissertation.
- Graf, Thomas & Connor Mayer. 2018. Sanskrit n-retroflexion is input-output tier-based strictly local. In *Proceedings of the 15th workshop on computational research in phonetics, phonology, and morphology*, 151–160. Brussels, Belgium: Association for Computational Linguistics. DOI: <https://doi.org/10.18653/v1/W18-5817>
- Heinz, Jeffrey, Chetan Rawal & Herbert G. Tanner. 2011. Tier-based strictly local constraints for phonology. In *Proceedings of the 49th annual meeting of the association for computational linguistics*, 58–64. Portland, Oregon, USA: Association for Computational Linguistics.

- Heinz, Jeffrey & James Rogers. 2013. Learning subregular classes of languages with factored deterministic automata. In Andras Kornai & Marco Kuhlmann (eds.), *Proceedings of the 13th meeting on the mathematics of language (MOL 13)*, 64–71. Sofia, Bulgaria: Association for Computational Linguistics.
- Heinz, Jeffrey & William Idsardi. 2013. What complexity differences reveal about domains in language. *Topics in Cognitive Science* 5(1). 111–131. DOI: <https://doi.org/10.1111/tops.12000>
- Hyman, Larry. 2009. How not to do typology: the case of pitch-accent. *Language Sciences*, 213–238. DOI: <https://doi.org/10.1016/j.langsci.2008.12.007>
- Hyman, Larry. 2011. Tone: Is it different? In John A. Goldsmith, Jason Riggle & Alan C. L. Yu (eds.), *The Blackwell handbook of phonological theory*, 197–238. West Sussex: Wiley-Blackwell. DOI: <https://doi.org/10.1002/9781444343069.ch7>
- Hyman, Larry. 2014a. How autosegmental is phonology? *The Linguistic Review* 31. 363–400. DOI: <https://doi.org/10.1515/tlr-2014-0004>
- Hyman, Larry & Francis X. Katamba. 2010. Tone, syntax and prosodic domains in Luganda. In Laura Downing, Annie Riailand, Jean-Marc Beltzung, Sophie Manus, Cédric Patin & Kristina Riedel (eds.), *Papers from the workshop on Bantu relative clauses (ZAS Papers in Linguistics)* 53. 69–98. ZAS Berlin. DOI: <https://doi.org/10.21248/zaspil.53.2010.393>
- Hyman, Larry M. 2014b. How autosegmental is phonology? *The Linguistic Review* 31(2). 363–400. DOI: <https://doi.org/10.1515/tlr-2014-0004>
- Jardine, Adam. 2016a. Computationally, tone is different. *Phonology* 33. 247–283. DOI: <https://doi.org/10.1017/S0952675716000129>
- Jardine, Adam. 2016b. *Locality and non-linear representations in tonal phonology*: University of Delaware dissertation.
- Jardine, Adam. 2017. The local nature of tone-association patterns. *Phonology* 34. 385–405. DOI: <https://doi.org/10.1017/S0952675717000197>
- Jardine, Adam. 2019. The expressivity of autosegmental grammars. *Journal of Logic, Language, and Information* 28. 9–54. DOI: <https://doi.org/10.1007/s10849-018-9270-x>
- Jardine, Adam. 2020. Melody learning and long-distance phonotactics in tone. *Natural Language and Linguistic Theory* 38. 1145–1195. DOI: <https://doi.org/10.1007/s11049-020-09466-y>
- Jardine, Adam & Jeffrey Heinz. 2016. Markedness constraints are negative: an autosegmental constraint definition language. In Ksenia Ershova, Joshua Falk, Jeffrey Geiger, Zachary Hebert, Robert E. Lewis Jr., Patrick Munoz, Jacob B. Phillips & Betsy Pillion (eds.), *Proceedings of the 51st annual meeting of the Chicago linguistic society*, 301–315.
- Kager, René. 1999. *Optimality theory*. Cambridge: Cambridge University Press. DOI: <https://doi.org/10.1017/CBO9780511812408>
- Karttunen, Lauri. 1998. The proper treatment of optimality in computational phonology. In *Proceedings of the international workshop on finite state methods in natural language processing*, 1–12. Ankara, Turkey: Association for Computational Linguistics. DOI: <https://doi.org/10.3115/1611533.1611534>
- Kenstowicz, Michael & Charles Kisseberth. 1990. Chizigula tonology: the word and beyond. In Sharon Inkelas & Draga Zec (eds.), *The phonology–syntax connection*, 163–194. Chicago: the University of Chicago Press.
- Kisseberth, Charles & David Odden. 2003. Tone. In Derek Nurse & Gérard Philippson (eds.), *The Bantu languages*, New York: Routledge.
- Koser, Nate & Adam Jardine. 2020. The computational nature of stress assignment. In Hyunah Baek, Chikako Takahashi & Alex Hong-Lun Yeung (eds.), *Proceedings of the 2019 annual meeting on phonology*. Linguistic Society of America. DOI: <https://doi.org/10.3765/amp.v8i0.4676>
- Koser, Nathan, Christopher Oakden & Adam Jardine. 2019. Tone association and output in non-linear structures. In Katherine Hout, Anna Mai, Adam McCollum, Sharon Rose & Matthew Zaslansky (eds.), *Supplemental proceedings of the 2018 annual meeting on phonology*. Linguistic Society of America. DOI: <https://doi.org/10.3765/amp.v7i0.4476>
- Leben, William R. 2006. Rethinking autosegmental phonology. In John Mugane (ed.), *Selected proceedings of the 35th annual conference on African linguistics*, 1–9. Somerville, MA: Cascadilla Proceedings Project.
- Levergood, Barbara. 1987. *Topics in Arusa phonology and morphology*: University of Texas dissertation.
- Lind, Douglas & Brian Marcus. 1995. *Symbolic dynamics and coding*. Cambridge: Cambridge University Press. DOI: <https://doi.org/10.1017/CBO9780511626302>
- McCarthy, John. 1996. Remarks on phonological opacity in Optimality Theory. In *Studies in Afroasiatic grammar: Papers from the second conference on Afroasiatic linguistics, Sophia Antipolis, 1994*, 215–243. The Hague: Holland Academic Graphics.
- McCarthy, John. 2010. Autosegmental spreading in Optimality Theory. In John A. Goldsmith, Elizabeth Hume & W. Leo Wezels (eds.), *Tones and features: Phonetic and phonological perspectives*, 195–222. Berlin/Boston: De Gruyter Mouton. DOI: <https://doi.org/10.1515/9783110246223.195>
- McCarthy, John & Alan Prince. 1993. Generalized alignment. In Geert Booij & Jaap van Marle (eds.), *Yearbook of morphology*, 79–153. Dordrecht: Kluwer. DOI: [https://doi.org/10.1007/978-94-017-3712-8\\_4](https://doi.org/10.1007/978-94-017-3712-8_4)



- McCarthy, John & Alan Prince. 1995. Faithfulness and reduplicative identity. In Jill Beckman, Laura Walsh Dickey & Suzanne Urbanczyk (eds.), *Papers in optimality theory* (University of Massachusetts Occasional Papers in Linguistics 18), 249–384. University of Massachusetts.
- McCarthy, John J. 2003. OT constraints are categorical. *Phonology* 20. 75–138. DOI: <https://doi.org/10.1017/S0952675703004470>
- McMullin, Kevin & Gunnar Ólafur Hansson. 2016. Long-distance phonotactics as tierbased strictly 2-local languages. In Adam Albright & Michelle A. Fullwood (eds.), *Proceedings of the annual meeting on phonology 2015*. Linguistic Society of America. DOI: <https://doi.org/10.3765/amp.v2i0.3750>
- Myers, Scott. 1997. OCP effects in Optimality Theory. *Natural Language & Linguistic Theory* 15(4). 847–892. DOI: <https://doi.org/10.1023/A:1005875608905>
- Odden, David. 1982. Tonal phenomena in Kishambaa. *Studies in African Linguistics* 13(2). 177–208.
- Odden, David. 1986. On the role of the Obligatory Contour Principle in phonological theory. *Language* 62(2). 353–383. DOI: <https://doi.org/10.2307/414677>
- Odden, David. 1994. Adjacency parameters in phonology. *Language* 70(2). 289–330. DOI: <https://doi.org/10.2307/415830>
- Odden, David. 1995. Tone: African languages. In John Goldsmith (ed.), *The handbook of phonological theory*, 444–475. Oxford: Blackwell.
- Odden, David. 2001. Tone shift and spread in Taita I. *Studies in African Linguistics* 30(1). 76–110.
- Odden, David. 2011. Features impinging on tone. In John Goldsmith, Elizabeth Hume & Leo Wetzels (eds.), *Tones and features*, 81–107. Berlin: De Gruyter. DOI: <https://doi.org/10.1515/9783110246223.81>
- Prince, Alan & Paul Smolensky. 1993. Optimality Theory: Constraint interaction in generative grammar. *Rutgers University Center for Cognitive Science Technical Report 2*.
- Prince, Alan & Paul Smolensky. 2004. *Optimality theory: Constraint interaction in generative grammar*. Oxford: Blackwell. DOI: <https://doi.org/10.1002/9780470759400>
- Rogers, James. 1998. *A descriptive approach to language-theoretic complexity*. Chicago: Center for the Study of Language and Information.
- Rogers, James & Geoffrey Pullum. 2011. Aural pattern recognition experiments and the subregular hierarchy. *Journal of Logic, Language and Information* 20. 329–342. DOI: <https://doi.org/10.1007/s10849-011-9140-2>
- Sande, Hannah, Peter Jenks & Sharon Inkelas. 2020. Cophonologies by ph(r)ase. *Natural Language & Linguistic Theory* 38. 1211–1261. DOI: <https://doi.org/10.1007/s11049-020-09467-x>
- Schadeberg, Thilo. 1979. Über die Töne der verbalen Formen im Rimi [On the tone of verbal forms in Rimi]. *Afrika und Übersee* 57. 288–313.
- Shih, Stephanie & Sharon Inkelas. 2019. Autosegmental aims in surface optimizing phonology. *Linguistic Inquiry* 50(1). 137–196. DOI: [https://doi.org/10.1162/ling\\_a\\_00304](https://doi.org/10.1162/ling_a_00304)
- Sibanda, Galen. 2004. *Verbal phonology and morphology of Ndebele*: UC Berkeley dissertation.
- Strother-Garcia, Kristina. 2019. *Using model theory in phonology: A novel characterization of syllable structure and syllabification*: University of Delaware dissertation.
- Vaysse, Odile. 1986. Addition molle et fonctions *p*-locales [Soft addition and *p*-local functions]. *Semigroup Forum* 34. 157–175. DOI: <https://doi.org/10.1007/BF02573160>
- Williams, Edwin S. 1976. Underlying tone in Margi and Igbo. *Linguistic Inquiry* 7(3). 463–484.
- Yip, Moira. 2002. *Tone*. Cambridge: Cambridge University Press.
- Zoll, Cheryl. 2003. Optimal tone mapping. *Linguistic Inquiry* 34(2). 225–268. DOI: <https://doi.org/10.1162/002438903321663398>

## TO CITE THIS ARTICLE:

Chandlee, Jane and Adam Jardine. 2021. Input and output locality and representation. *Glossa: a journal of general linguistics* 6(1): 43. 1–27. DOI: <https://doi.org/10.5334/gjgl.1423>

Submitted: 01 September 2020

Accepted: 22 February 2021

Published: 08 April 2021

## COPYRIGHT:

© 2021 The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CC-BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited. See <http://creativecommons.org/licenses/by/4.0/>.

*Glossa: a journal of general linguistics* is a peer-reviewed open access journal published by Ubiquity Press.