Noname manuscript No.
(will be inserted by the editor)

Melody learning and long-distance phonotactics in tone

Adam Jardine

the date of receipt and acceptance should be inserted later

Abstract This paper presents evidence that tone well-formedness patterns share a property of *melody-locality*, and shows how patterns with this property can be learned. Essentially, a melody-local pattern is one in which constraints over an autosegmental melody operate independently of constraints over the string of tone-bearing units. This includes a range of local tone patterns, long-distance tone-patterns, and their interactions. These results are obtained from the perspective of formal language theory and grammatical inference, which focus on the structural properties of patterns, but the implications extend to other learning frameworks. In particular, a melody-local learner can induce attested tone patterns that cannot be learned by the tier projection learners that have formed the basis of work on learning long-distance phonology. Thus, melody-local learning is a necessary property for learning tone. It is also shown how melody-local learners are more restrictive than learning directly over autosegmental representations
Keywords: Tone, learnability, computational phonology, representation

1 Introduction

1.1 The proposal

Long-distance generalizations pose a challenge for learning, because a learner must discover dependencies that may hold over an unbounded distance. Work on learning phonotactics has made gains towards solving this problem by adopting representational mechanisms that allow a learner to discover long-distance dependencies. Specifically, previously proposed phonotactic learners have either made reference to precedence relations (Heinz, 2010a), tier projection (Hayes and Wilson, 2008; Goldsmith and Riggle, 2012; Jardine and Heinz, 2016; McMullin and Hansson, 2015; Jardine and McMullin, 2017; Gallagher and Wilson, 2018), or a generalization of the two (Graf, 2017).

The author would like to thank Jane Chandlee, Jeffrey Heinz, Arto Anttila, and three anonymous reviewers for their thoughtful comments.
Manuscript submitted to *Natural Language & Linguistic Theory*, revised January 20, 2020.

Department of Linguistics, Rutgers University
E-mail: adam.jardine@rutgers.edu

However, the specific problem of learning tonal patterns has not been yet studied in this way. This paper fills this gap by studying the learnability of long-distance phonotactic patterns in tone. This reveals that previously proposed representational mechanisms for learning phonotactic patterns are insufficient for learning the types of patterns that are found in tone. This article instead argues that long-distance phonotactic patterns in tone must be learned using a distinct notion of a *melody*, or string of unique components of surface tone. Evidence is given below that tone patterns share a property of *melody-locality*, meaning that they are describable by the intersection of a set of local constraints over strings of tone-bearing units (TBUs) and a set of local constraints over a melody.

For example, in Prinmi (Tibeto-Burman, China; Ding, 2006; Hyman, 2009), words must have exactly one span of H tones, which can be either one (1a) or two (1b) TBUs long. In (1) below, schematic representations of the forms as strings of H and L TBUs are given on the left. In the full transcription, as throughout this paper, an acute accent on the vowel indicates a H toned TBU; all other TBUs are low (L).

(1) Prinmi (Ding, 2006, p. 14)

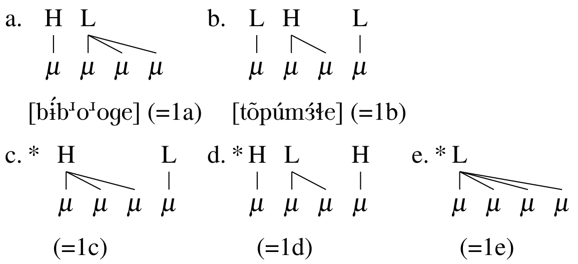
a.	HLLL	[bɨ́bʴobʴoge]	‘as for roasted flour’
b.	LHHL	[tõpúmɜ́ɬe]	‘donkey tail’
c.	*HHHL		
d.	*HLLH		
e.	*LLLL		

Spans of more than two consecutive H-toned TBUs are illicit (1c), as are two distinct spans of H-toned moras (1d). The Prinmi pattern can thus be captured by the satisfaction of three constraints: one banning three consecutive H-toned TBUs (2a), one banning HLH sequences in the melody (2b; this essentially functions like Yip 2002’s *Trough), and one banning melodies with a single L (2c; this functions like Hyman 2009’s Obligatoriness constraint for H tones).

(2)

a.	*HHH (TBU string):	No HHH sequence in the string of TBUs.
b.	*HLH (melody):	No HLH sequence in the melody.
c.	*#L# (melody):	No all-L melody.

Here, ‘melody’ is used in the sense of a string of tones on an independent tonal tier (a la Leben, 1973; Williams, 1976; Goldsmith, 1976). As the autosegmental representations (ARs) in (3) of the forms in (1) show, the melody of the ill-formed word *HLLH in (1d) contains a HLH sequence (3d). Likewise, the melody of *LLLL in (1e) is, when boundaries are made explicit, #L# (3e). In contrast, the ARs of HLLL and LHHL do not contain either of these substrings in their melodies (3a, b).

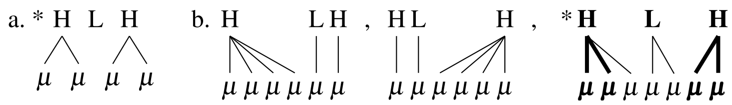
(3)


The surface pattern in Prinmi is thus describable by taking the intersection of the set of forms that satisfy (2a) and the set of forms that satisfy (2b) and (2c). As seen in the examples in 3, this assumes melodies that obey the Obligatory Contour Principle (OCP; Leben, 1973; Odden, 1986; Myers, 1997). As discussed below, this allows local constraints over melodies to capture long-distance generalizations such as “there cannot be two distinct H spans in a word.”

The first major goal of this paper is to define melody-local patterns as those that, like Prinmi, are describable with the intersection of constraints on sequences of consecutive TBUs and constraints on sequences of tones in the melody. Crucially, these grammars operate not over ARs directly, but on strings of TBUs and strings of melodies. This thus forms a strong hypothesis about tone: that constraints either refer to the melody or to the string of TBUs, but cannot refer to arbitrary associations between them.

For example, melody-local grammars allow a melody constraint as in (2b) *HLH that is violated when an HLH sequence appears in the melody, but do not allow a constraint like as in (4a), which is violated only when *both* an HLH sequence occurs in the melody *and* both Hs are doubly-associated.

(4) A constraint that is impossible under the melody-local hypothesis



As shown in (4b), such a constraint is evaluated as a kind of non-local conjunction: it allows one H or the other to spread, but not both at once. As discussed in §6.1, such a constraint excluded by the melody-local hypothesis, as (4a) combines melody information and local information in a single constraint. This prediction is borne out by the typology of tone, as this constraint is unattested in tone.

Thus, the hypothesis that tone is melody-local is a predictive one that makes principled distinctions between attested long-distance patterns and logically possible, yet unattested patterns. These predictions are largely borne out because, as detailed below, local and long-distance well-formedness tone patterns are predominantly melody-local. This includes patterns like Prinmi, in which local and long-distance generalizations interact. A somewhat problematic case, in Karanga Shona, is also discussed, but it is also largely melody-local.

1.2 Contrast with existing models of long-distance learning

That tone patterns are melody-local has consequences for learnability. The other major goal of this paper is to show that melody-local patterns can be efficently learned. This is contrasted with previously proposed phonotactic learners, which cannot learn the attested tone patterns discussed in this paper. The conclusion, then, is that melody-localily is a necessary property for learning tone.

While the melody described above is a *tier* in Goldsmith (1976)’s original conception of the term, it is demonstrably distinct from the *tier projection* representations used throughout the literature on learning phonology. Tier projections are used by learners based in information theory (Goldsmith and Riggle, 2012), maximum entropy (Hayes and Wilson, 2008; Gallagher and Wilson, 2018), and grammatical inference (McMullin and Hansson, 2015; Jardine and Heinz, 2016). However, tier projections cannot capture the full range of tone patterns described here, in particular because they cannot distinguish between units that are adjacent and those that are non-adjacent in the surface string of TBUs.

Prinmi is a useful illustration. Tier projection models capture long-distance patterns by 'ignoring' some subset of units in the surface string (Heinz et al., 2011). In Prinmi, the non-local constraint against multiple H-tone spans would have to be captured by projecting a tier of H-toned TBUs, ignoring L-toned TBUs. This is depicted below in (5).

(5) a. * H   H   b.   H H
        ↑   ↑   ↑ ↑
        H L L H (=1d)   L H H L (=1b)

Forms like (1d) with multiple H-tone spans could then be ruled out by a *HH constraint operating on the tier projection. However, since Prinmi allows spreading of H-tone up to two TBUs, it is possible for there to be more than one H-toned TBU to appear in the string, as shown above in (5b). Thus, the *HH constraint would also incorrectly rule out well-formed words like (1b). Thus, tier projection cannot distinguish between well- and ill-formed words in a pattern like Prinmi's, because it cannot distinguish between consecutive and non-consecutive H-toned TBUs. As discussed in §5, the same is true for the learners based on precedence (Heinz, 2010a) or intervals (Graf, 2017). This is not to say that tier projections are not necessary or useful in phonology. Indeed, TBUs are a projection of sorts from the surface string of segmental information. However, the goal of this article is to show that tier projections alone are not *sufficient* for learning tone.

1.3 Scope

Thus, melody-local learning is a novel, and necessary, step towards understanding how tone is learned. In a way, this result confirms the utility of the autosegmental concepts of an independent melody and the OCP. The novel contribution is to rigorously apply these concepts using formal language theory, the formal study of patterns (Hopcroft et al., 2006), and grammatical inference, which is concerned with the inductive learning of patterns from finite sets of examples (de la Higuera, 2010; Heinz et al., 2016). This follows previous formal work on ARs (Bird and Ellison, 1994; Kornai, 1995; Jardine, 2017) and expands on it with a concrete learnability result.

The value in formal language-theoretic descriptions is that they tell us about the structural properties of a pattern. For example, the *HHH constraint in (2a) is a *strictly local* constraint (McNaughton and Papert, 1971; Rogers and Pullum, 2011), because it refers to sequences of adjacent units in a string. We can also prove when a pattern cannot be captured by tier projection. This tells us when a learner that operates over particular kinds of constraints—e.g. strictly local or based on tier projection—are guaranteed to fail to learn the pattern, no matter how much data it is exposed to.

Thus, this paper abstracts away from the problems of learning in the face of exceptions, or learning gradient generalizations, which is the focus of much work on phonotactic learning (e.g., Hayes and Wilson, 2008; Gallagher and Wilson, 2018). This is not to say that these are not important problems. However, they are a separate issue from identifying the right kinds of representations over which learners operate. A learner paying attention to strictly local information will never learn a long-distance pattern, no matter how sophisticated its statistics. Similarly, a tier projection learner, stochastic or no, will never learn a pattern that cannot be represented with tier projection. Thus, in order to clearly focus on these structural properties, this paper adopts a categorical learning approach.

However, for the reasons just stated, the results of this paper bear directly on probabilistic learners. Stochastic learning often builds on results from the categorical perspective (de la Higuera, 2010; Heinz and Rogers, 2013). Indeed, maximum entropy learners for long-distance phonotactics (Hayes and Wilson, 2008; Gallagher and Wilson, 2018; Gouskova and Gallagher, pear) share the same basic tier projection structure as formal language-theoretic characterizations of long-distance phonotactics (Heinz et al., 2011). Thus, they will not be able to learn the full range of tone patterns discussed in this paper. Section 5.3 sketches how a maximum entropy model can build on the results of this paper.

Finally, as discussed in more detail below, this paper also abstracts away from the problem of learning alternations, and instead focuses on static well-formedness generalizations. Tonal phonologies do not only consist of well-formedness patterns, and very often are analyzed terms of a complex of interacting processes mapping underlying forms to surface forms. The learning of such a complex system is an enormous task, and so this paper adopts a piecemeal approach by first focusing on phonotactics. There is support for such an approach. Empirically, phonotactic knowledge has been shown to aid learning of alternations (Pater and Tessier, 2003). Theoretically, proposals for learning alternations can be based directly on results on the learnability of static generalizations (Chandlee and Heinz, 2018). Thus, following, for example, Hayes and Wilson (2008) and Heinz (2010a), this paper views learning phonotactics as an important step towards solving the more general problem of learning entire phonologies.

1.4 Outline of the paper

This paper is structured as follows. §2 defines a property of melody-locality and §3 surveys a range of local and non-local tone patterns that share this property. §4 demonstrates a learner for melody-local patterns, and §5 contrasts this learner with established learners based on precedence and tier projection. §6 discusses representational issues and future work, and §7 concludes. An Appendix collects formal definitions and proofs for concepts used throughout the paper.

2 The proposal

This paper proposes a formal characterization of tonal well-formedness that 1) checks local constraints over the tone specifications of TBUs and 2) checks local constraints over the tonal melodies. This forms a strong hypothesis that constraints can refer either to the melody or to the string of TBUs, but no single constraint can refer to both at the same time. This property of *melody-locality* is obtained through studying phonological well-formedness through the lens of formal language theory.

2.1 What is a pattern?

The term ‘pattern’ in this paper refers to a well-formedness generalization holding over output representations. For example, in Kagoshima Japanese (Japonic, Japan), words have a H tone either on the penultimate or final syllable (Hirayama, 1951; Haraguchi, 1977; Kubozono, 2012). Note that this can create contrasts, as in the minimal pair (6a) [hána] ‘nose’ and (6e) [haná] ‘flower’.

(6) Kagoshima Japanese (Haraguchi, 1977, p. 195)

a.	hána	'nose'	HL
b.	sakúra	'cherry blossom'	LHL
c.	kagaríbi	'watch fire'	LLHL
d.	kagaribí-ga	'watch fire' + NOM	LLLHL
e.	haná	'flower'	LH
f.	usagí	'rabbit'	LLH
g.	irogamí	'color paper'	LLLH
h.	irogami-gá	'color paper' + NOM	LLLLH

This generalization is a *set* of well-formed strings of H-toned and L-toned TBUs.[1] Let us call this set L_{KJ}.

(7) $L_{\mathrm{KJ}} = \{$H, HL, LHL, LLHL, LLLHL, ..., LH, LLH, LLLH, LLLLH, ...$\}$

This set includes all and only the strings of Hs and Ls such that a single H falls on either the ultimate or penultimate position. Strings such as *LHLLL, which do not conform to the generalization, are not members of the set.

Note that the ellipses indicate that L_{KJ} is an infinite set. Thus, for example, LLLLLLLHL is in L_{KJ}, even if a word of such length may not exist. However, this models the fact that speakers of Kagoshima Japanese *would* find such a word well-formed, if it were created through morphological processes or borrowing. Indeed, borrowing of English words has created long words such as [makudonarúdo] 'McDonald's', which nevertheless conform to the generalization (Kubozono, 2012).

Tone presents an interesting challenge in isolating well-formedness generalizations. Many existing analyses of tone systems are heavily derivational, with multiple levels of interacting processes and well-formedness generalizations. A prime example discussed below is unbounded tone plateauing in Luganda (Hyman and Katamba, 2010), in which two underlying H tones merge to create a single H-toned plateau. In some cases, the superimposition of phrasal boundary tones creates forms that superficially violate this generalization (see Hyman and Katamba, 2010, p. 72). However, this does not mean that the plateauing generalization is not psychologically real, as it is still plays a crucial role in the overall phonology of the language (Hyman and Katamba, 1993, 2010).[2] This paper thus takes examples from analyses for which there is a clear well-formedness generalization that holds at some significant level posited by the analyst (as with Kagoshima Japanese and unbounded tone plateauing in Luganda, usually the lexical or early phrasal phonology).

By doing so, we can focus on how generalizations like unbounded tone plateauing can be learned, abstracting away from their position in a larger phonology. This is not unlike isolating a particular process to study how it would be formalized with a single rule, or fragment of OT ranking. In terms of learning, this is somewhat of an idealization, as it assumes the learner has already identified the right domain and level of derivation for application of the process. However, this idealization allows us to focus directly on the problem: *even given all*

[1] Pierrehumbert and Beckman (1988) propose an underspecification-based model for (Tokyo) Japanese. A partial string-based representation of this model would instead represent the L-toned TBUs as ∅ TBUs, indicating that they are not linked to any tone, and H, which indicates the TBU linked to the H tone. How we might fully capture the spirit of underspecification models via string representations is briefly discussed in §6.3.

[2] Hyman and Katamba (2010)'s account of Luganda is derivational, explicitly distinguishing between lexical and post-lexical levels of the phonology. This paper follows this assumption that the phonology can be organized into distinct sub-phonologies, although it abstracts away from the details of how these are organized. In a constraint-based framework, this would require a stratally organized grammar (Kiparsky, 2000; Bermúdez-Otero, 2017).

of this information, previous learning models cannot learn all of the tone well-formedness generalizations surveyed in the following sections.

2.2 What is a local pattern?

Armed with a notion of well-formedness generalizations as sets, we can rigorously define a notion of *locality*. The term 'local' is oft-used but can have various meanings, sometimes meaning strict adjacency (Gafos, 1996; Chiośain and Padgett, 2001) or adjacency on a tier (Odden, 1994). This paper adopts the following definition, based on the *strictly local* class of formal languages (McNaughton and Papert, 1971; Rogers and Pullum, 2011).

(8) A *local grammar* is a finite set S of forbidden *substrings*. Let $L(S)$ be the set of strings not containing any element in S; we say that S *describes* $L(S)$. A pattern L is *local* iff there is a finite set S of forbidden substrings that describes it.

A *substring* is simply a sequence of symbols in a string (including itself); the substrings of the string LLH are L, H, LL, LH, and LLH. For example, the Kagoshima Japanese pattern L_{KJ} is local because it is describable with the following local grammar S_{KJ}, which is a set of forbidden substrings of length at most 3. Here, # is an extra symbol indicating a word boundary.

(9) $S_{\text{KJ}} = \{\#\text{L}\#, \text{HH}, \text{HLL}, \text{HLH}, \text{LL}\#\}$

The grammar S_{KJ} can be interpreted as follows: #L# bans monosyllabic L words; HH bans words with two adjacent H tones; HLL and HLH ban words in which a H appears anywhere to the left of penultimate position; and LL# requires that either the penult or final syllable is H (recall that they both cannot be H, as HH has been banned). The set L_{KJ} is thus exactly the set of strings of Ls and Hs containing none of the substrings in S_{KJ}.

Forbidden substring constraints are computationally very simple (Rogers and Pullum, 2011; Rogers et al., 2013) and, as to be shown in more detail in §4, are efficiently learnable (García et al., 1990; Heinz, 2010a). This stems from their cognitive interpretation. For a set S of forbidden substrings of at most length k, we can check to see if a string is well-formed with respect to S simply by scanning through the string with a window of size k. This is illustrated for L_{KJ} and its grammar S_{KJ} in (10) below.

(10) a. $\boxed{\#\,\text{L}\,\text{L}}\,\text{H}\,\text{L}\,\#$ $\#\,\boxed{\text{L}\,\text{L}\,\text{H}}\,\text{L}\,\#$ $\#\,\text{L}\,\boxed{\text{L}\,\text{H}\,\text{L}}\,\#$ $\#\,\text{L}\,\text{L}\,\boxed{\text{H}\,\text{L}\,\#}$
✓ #LL   ✓ LLH   ✓ LHL   ✓ HL#

b. $\boxed{\#\,\text{L}\,\text{H}}\,\text{L}\,\text{L}\,\#$ $\#\,\boxed{\text{L}\,\text{H}\,\text{L}}\,\text{L}\,\#$ $\#\,\text{L}\,\boxed{\text{H}\,\text{L}\,\text{L}}\,\#$
✓ #LH   ✓ LHL   ✗ HLL

Both (10a) and (b) show a scanning window of length 3 (indicated by a box) move sequentially through the strings #LLHL# and #LHLL#. Below each box is the substring of length 3 that currently appears in the window. If *at any time* one of the substrings in S_{KJ} appears in the window, the form is judged by the procedure to be ungrammatical. In (10a), the scanner moves all the way through the string without encountering a substring from S_{KJ}, so the form is judged to be grammatical. In (10b), the scanner encounters the forbidden substring HLL$\in S_{\text{KJ}}$ on the third step, and so *LHLL is correctly judged to be ungrammatical. Thus, patterns that can be described with forbidden substring grammars are *local* because their well-formedness generalizations depend entirely on information that can be detected in some fixed window.

Local patterns can thus be induced by a learner that scans through the input in this manner, remembering the sequences it has seen in its fixed window. This procedure is explained in more detail in §4.2. While we have defined this in terms of formal language theory, this same basic structure is used widely in natural language processing and learners in theoretical phonology; *n*-gram models (Jurafsky and Martin, 2009) and the MaxEnt models in Hayes and Wilson (2008) (with the exception of that for Shona) are local in this sense.

Kagoshima Japanese is just one example of a local pattern, but constraints that are local in this way are common in tone. Constraints referring to some position near the word edge (like that in Kagoshima Japanese), bounded spreading (Hyman, 2007, p.7), local interpretations of the Obligatory Contour Principle that forbid H tones on adjacent TBUs (such as proposed by Myers 1997, pp. 847–8, for Shona; see also *CLASH in Zoll 2003, pp. 239–40, for Kukya), a local *HLH constraint that forbids a single L-toned TBU in between two Hs (e.g., Kihunde; Goldsmith, 1990, p. 36), and 'edge-based' constraints that restrict contours or plateaus to a word edge (e.g. as in Mende; Leben, 1973, pp. 64-5), all conform to this definition of locality.

2.3 Non-local patterns

However, there are also well-formedness generalizations in tone that are *non*-local according to this definition. One common non-local pattern is *unbounded tone plateauing* (UTP) (Kisseberth and Odden, 2003; Hyman, 2011; Jardine, 2016), in which surface well-formedness dictates that only one plateau, or unbroken stretch, of high-toned TBUs is allowed in a domain. The following examples are from Luganda (Bantu, Uganda; Hyman and Katamba, 2010; Hyman, 2011). Underlying forms are given on the left. The output forms here are what Hyman and Katamba (2010) call "intermediate forms"—the output of the phonology before the imposition of phrasal boundary tones.

(11) Luganda (Hyman and Katamba, 2010, pp. 71, 78)

a.	/kitabo/	'book'	kitabo	LLL
b.	/mutéma/	'chopper'	mutéma	LHL
c.	/kisikí/	'log'	kisikí[3]	LLH
d.	/mutéma+bisikí/	'log chopper'	mutémá+bísíkí	LHHHHH
e.			*mutéma+bisikí	*LHLLLH

Examples (11a) through (c) show that forms with no underlying Hs or a single H surface faithfully (again, modulo phrasal boundary tones). However, a form with more than one H tone, as in the compound in (11d), surfaces with a plateau of H-toned TBUs in between the first and last H—no matter how far apart.

Hyman and Katamba (2010) characterize UTP derivationally as a process which merges two H tones together. In terms of surface well-formedness, this pattern can be characterized as the result of a long-distance *HLH (or *TROUGH, in Yip (2002)'s terms) constraint which bans *any* two H-toned TBUs separated by any number of L-toned TBUs (Hyman, 2011). The resulting well-formedness generalization is that forms produced by UTP contain at most one

[3] On the surface underlying final H tones are pronounced with a falling tone; thus /kisikí/ 'log' is pronounced [kisikî]. However, in order to focus on the long-distance aspects of this pattern, this discussion will for now abstract away from contours and thus they will not be transcribed. For discussion on how contours can be straightforwardly incorporated into the proposal, see §6.2.

plateau of H-toned TBUs. This can be formalized with the set L_{UTP} of strings that satisfy this constraint are given in (12).

(12) $L_{\mathrm{UTP}} = \{$ L, H, LL, LH, HL, HH, LLL, LLH, LHL, LHH, HLL, HHL, HHH, LLLL, LLLH, LLHL, LLHH, LHLL, LHHL, LHHH, HLLL, HHLL, HHHL, ... $\}$

In other words, L_{UTP} is exactly the set of strings that do not contain any sequence HL^{n}H, where L^n indicates a sequence of n L-toned TBUs. This is non-local because there is no scanner window of fixed length that can distinguish between well-and ill-formed strings. This is illustrated diagrammatically in (13).

(13) a. # L $\boxed{\text{L L L} \ldots \text{ L}}$ H #
✓ LLL...L

b. * # H $\boxed{\text{L L L} \ldots \text{ L}}$ H #
✓ LLL...L

The well-formed string in (13a) has some arbitrary number of Ls followed by a H. A fixed scanner window must accept this sequence of Ls, as the string contains a single H-toned span (of exactly one H) and is thus well-formed. However, our scanner cannot distinguish this sequence of Ls from the one in the ill-formed string in (13b). The reason for this is that, because it is checking substrings of some bound length, it cannot ‘remember’ that it has not seen a H to the left in (13a), or that it has seen a H to the left in (13b). Thus it incorrectly will judge both strings to be well-formed. Importantly, increasing the length of substrings that the scanner checks does not help—there will always be some pair of strings like (13a) and (13b) that it cannot distinguish.[4] This is an intuitive explanation of why L_{UTP} is not local; a formal characterization of strictly local formal languages given in the Appendix allows us to *prove* that a pattern is not local (Rogers and Pullum, 2011).

Importantly, this means that there is no local learner that can learn the UTP pattern. This is because it is outside of the hypothesis space of the learner: as a local learner can only use information it has seen in the fixed space of the scanning window, it can only posit grammars based on this information. In the specific paradigm considered in this paper, this means that a local learner can only posit a forbidden substring grammar, such as S_{KJ} in (9) in the preceding section. Thus, it can never posit a grammar that forbids HL^{n}H for any n. This is also true for statistically-based local learners: the information they keep track of is also restricted to some fixed window in the same way. Thus, no matter how sophisticated its statistics, or how much data it is given, a local learner cannot learn a long-distance pattern such as UTP.

2.4 Melody constraints

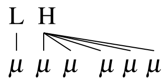
Thus, in order to capture long-distance patterns, a learner must be sensitive to some other information in input strings. A common approach in phonological learners is to use a function that projects a tier, eliminating irrelevant information in the string, and then performing local learning over that tier (Hayes and Wilson, 2008; Heinz et al., 2011; Goldsmith and

[4] Recall our assumption that generalizations hold for strings of arbitrary length. Thus, property of locality is about the phonological pattern itself, independent of generalizations about word length or of constraints on performance or processing.

Riggle, 2012). However, as detailed in §5, this is insufficient for learning long-distance tone patterns. Instead, this paper shows that learning long-distance patterns in tone, as well as their interactions with local patterns, requires representing local constraints over *melodies*.

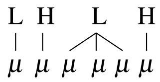
Here, melody is used in its original sense of a string of tonal autosegments (Leben, 1973; Williams, 1976; Goldsmith, 1976). (This is in contrast with other uses of the term melody, as in strings of consonant or vowel phonemes in non-concatenative morphology, i.e. in McCarthy 1985.) For example, the plateauing example from Luganda, (11c) [mutémá+bísíkí] 'log chopper,'would be represented autosegmentally as in (14a), and the ill-formed *[mutéma+bisikí] as in (14b).

(14) a.



[mutémá+bísíkí]
'log chopper' (=11d)

b. *



*[mutéma+bisikí]
(=11e)

We can then posit that in Luganda, ill-formed words contain a HLH sequence in the melody of their AR. This distinguishes (14a), whose melody is LH, from (14b), whose melody is *LHLH, which contains the illicit substring HLH. Note that this requires assuming the OCP—that is, that only a single L can intervene between two Hs on the melody.

The core idea of this paper is to posit local constraints directly over tonal melodies, instead of representing ARs in full. This melody can be extracted with a function operating on a surface string of TBUs, much in the same way as a tier-projection function operates. We can define a function $\mathtt{mldy}(w)$ that takes a string w of H and L-toned TBUs and replaces each *span* (or unbroken sequence) of Hs or Ls with a single H or L, in the order they appeared. This process is shown step-by-step in (15) for LHHHH (the string of TBUs corresponding to (11)d) [mutémá+bísíkí] 'log chopper') and LHLLH (corresponding to (11)e) *[mutéma+bisikí]).[5]

(15) a. $\mathtt{mldy}(\text{LHHHHH}) = \underset{\text{L}}{\boxed{\text{L}}}\underset{\text{H}}{\boxed{\text{H H H H H}}} = \text{LH}$

b. $\mathtt{mldy}(\text{LHLLLH}) = \underset{\text{L}}{\boxed{\text{L}}}\underset{\text{H}}{\boxed{\text{H}}}\underset{\text{L}}{\boxed{\text{L L L}}}\underset{\text{H}}{\boxed{\text{H}}} = \text{LHLH}$

In (15a), the string LHHHHH is composed of a span of a single L-toned TBU followed by a span of five H-toned TBUs.[6] The mldy function collapses these two spans into a string LH which represents a L span followed by a H span. Similarly, in (15b), the string LHLLLH is composed of a single L, a single H, a span of three Ls, and a H. Collapsing the span of three Ls into a single L, the function obtains LHLH. We have now generated exactly the melody strings from the ARs in (14a) and (14b), respectively.

We then posit a grammar M_{UTP} of forbidden substrings to be interpreted as holding over these melody strings.

[5] This function is defined formally in the Appendix. Interestingly, this function is *input strictly local*, a restrictive type of function that has been linked to phonological processes (Chandlee and Heinz, 2018; Chandlee et al., 2018). Thanks to Jane Chandlee and Jeff Heinz for pointing this out.

[6] This function is based on Jardine and Heinz (2015)'s concatenation operation for generating OCP-obeying ARs from strings. For now, we gloss over the treatment of contours, which can be straightforwardly dealt with but are not necessary to capture the tone patterns from §3. It will be shown in §6.2 how to adapt our melody function to incorporate contours.

(16) $M_{\text{UTP}} = \{\text{HLH}\}$

We use the 'M' notation to indicate forbidden substring grammars that operate over the strings generated by `mldy`. Given this, M_{UTP} will be violated by any string w of TBUs for which $\texttt{mldy}(w)$ contains an HLH sequence. Thus, for example, *LHLLLH is *not* in the set of strings described by M_{UTP}, because $\texttt{mldy}(\text{LHLLLH}) = \text{LHLH}$, which contains the substring HLH. However, LHHHHH *is* in the set of strings described by M_{UTP}, because $\texttt{mldy}(\text{LHHHHH}) = \text{LH}$ does not include the forbidden substring HLH.

In general, $\texttt{mldy}(w)$ for any string w with two Hs separated by at least one L will include the forbidden substring HLH, and thus will be excluded by M_{UTP}. Thus, the grammar M_{UTP} describes exactly the set L_{UTP} from §2.3 representing the long-distance UTP generalization. It has done this by implementing a local constraint against a HLH sequence *in the melody* of an AR. This reduction of a long-distance pattern to a local melody constraint depends on the assumption that, on the surface, in ARs the OCP merges all adjacent, like-toned TBUs. This and other representational assumptions will be discussed in more detail in §6.1.

Because the pattern is local when viewed over a melody, it is learnable with the same scanning procedure outlined in §2.2. This is demonstrated in detail in §4.3. First, however, the following sections show how these can be combined with local constraints to provide a learnable model of tonal well-formedness.

2.5 Melody-local grammars

As shown in the preceding section, tone includes both local and long-distance well-formedness generalizations. Thus, any theory of tonal well-formedness must be able to account for both. The proposal here is then that tonal well-formedness patterns can be captured by *melody-local* grammars that pair a forbidden substring grammar S operating over surface strings with a melody grammar M operating over the melodies of strings, as in (17).

(17) $G = (S, M)$

We define $L(G)$, the set described by G, as in (18).

(18) $L(G)$ is the set of all strings w such that
1. w contains none of the substrings in S, and
2. $\texttt{mldy}(w)$ contains none of the substrings in M.

Crucially, (18) is defined *conjunctively*; a string w satisfies the grammar if and only if it satisfies *both* (18a) and (18b).

We can then define a new version of locality, call it *melody-local*, parallel to the definition of locality in (8).

(19) A pattern is *melody-local* iff it is describable by a melody-local grammar.

For example, the Kagoshima Japanese pattern L_{KJ} and the UTP pattern L_{UTP} are both melody-local, as they can be described with the melody-local grammars below.[7]

(20) 1. $G_{\text{KJ}} = (S_{\text{KJ}} = \{\text{\#L\#}, \text{HH}, \text{HLL}, \text{HLH}, \text{LL\#}\}, M_{\text{KJ}} = \{\})$

[7] These are not the *only* way these patterns can be described with melody-local grammars; in fact the learning algorithm proposed in Sect. 4 will learn slightly different, though extensionally equivalent, grammars. For discussion see §§4.2 and 4.3.

2. $G_{\text{UTP}} = (S_{\text{UTP}} = \{\}, M_{\text{UTP}} = \{\text{HLH}\})$

In the melody-local grammar for a purely local patterns like L_{KJ}, M is empty. Likewise, for purely long-distance patterns with no local constraints, S is empty. Since (18) defines satisfaction of a melody-local grammar as containing no elements in either S or M, if either is empty, then it is always vacuously satisfied.

In cases in which both S and M are nonempty, the grammar describes a pattern in which a local and long-distance generalization interact. However, they can only interact conjunctively; strings must satisfy both the local constraint *and* the melody constraint. No single constraint can refer to both melody information and local information at the same time. We can thus take (19) to be a restrictive *hypothesis* about tone—that tonal well-formedness patterns must be melody-local. As the following section shows, this prediction is largely borne out by the kinds of long-distance patterns exhibited in tone.

3 Evidence for melody-locality in tone

This section applies the above-defined notion of melody-locality and shows that a sample of tonal well-formedness patterns exemplifiying major types of unbounded generalizations in tone are melody-local. A comprehensive review of tone is difficult, as there is a vast range of patterns in the literature (Yip, 2002; Hyman, 2011). However, as already noted, many tone patterns are purely local. Thus, this section will focus on the range of unbounded patterns that appear in tone, with a particular focus on patterns in which long-distance and local constraints interact. The following surveys obligatoriness in Chuave (Donohue, 1997), culminativity in Arigibi (Donohue, 1997) and Prinmi (Ding, 2006), and the blocking of H-spread in Bemba (Bickmore and Kula, 2013, 2015), and shows that all are melody-local. Additionally, a marginal case in Karanga Shona (Odden, 1982; Hewitt and Prince, 1989), which is melody-local save for one exception, is discussed. Regardless, as to be seen in §§4 and 5, these patterns make clear the necessity of considering melody-locality for a learning model of tone. Finally, as further evidence for melody-locality as a strong characterization of the structure of tone patterns, this section highlights some logically possible, yet unattested tone patterns that are not melody-local.

This paper adopts Hyman (2001)'s definition of a tone language as "one in which an indication of pitch enters into the lexical realization of at least some morphemes" (p. 1368). This definition includes patterns that are sometimes referred to as "pitch accent" (as in Kagoshima Japanese) but it is not clear that these patterns should be treated as distinct from other tone patterns (Hyman, 2009). This also holds for their formal properties: "pitch accent" systems and uncontroversial tone systems, at least the ones surveyed here (and in Jardine 2016, 2019), exhibit the same kinds of local and non-local patterns.

Finally, it is common for tone patterns to only apply to a particular domain. The following assumes that domain knowledge is pre-specified: what is the complexity of pattern L that occurs within a particular domain? Domain information can be included by directly encoding different types of boundaries in the representation; how this affects learning is a distinct, if important, learning problem (see Graf, 2017).

3.1 Obligatoriness in Chuave

UTP can be considered what Hyman (2009) calls a *culminativity* constraint: it forbids the existence of more than one H plateau in a domain. As Hyman discusses, another kind of unbounded pattern in tone is that of *obligatoriness*, where at least one tone must appear in a word. One example he gives is Chuave (Chimbu, Papua New Guinea; Donohue, 1997), in which every word must have at least one H-toned TBU. The relevant TBU is the mora. Donohue does not explicitly state at which level of the derivation this pattern holds, but one can infer from his description that it is true on the surface.

(21) Tone in Chuave (Donohue, 1997, p. 355)

a.	kán H	'stick'	e.	gíngódí HHH	'snore'	i.	kóiom HLL	'wing'
b.	gáán HH	'child'	f.	dénkábu HHL	'mosquito'	j.	komári LHL	'before'
c.	gáam HL	'skim'	g.	énugú HLH	'smoke'	k.	koiyóm LLH	'navel'
d.	kubá LH	'bamboo'	h.	amámó LHH	'k. o. yam'			

Words of the shape *L, *LL, *LLL, etc., are ill-formed in Chuave—in other words, a H tone is obligatory. The well-formedess condition in Chuave is represented in (22) as the set L_{Ch} of strings with at least one H.

(22) $L_{\mathrm{Ch}} = \{$H, HL, LH, HH, LLH, LHL, LHH, HLL, HLH, HHH, LLLH, ...$\}$

Like UTP, this is non-local because there is no window of fixed length that can distinguish all and only the strings with no H.

However, for any string consisting of exactly one L tone span, the `mldy` function will condense it into a single string L: $\mathtt{mldy}(\mathrm{L}) = \mathtt{mldy}(\mathrm{LL}) = \mathtt{mldy}(\mathrm{LLL}) = \mathrm{L}$. We can then ban all such strings by positing a set $M_{\mathrm{Ch}} = \{\#\mathrm{L}\#\}$ consisting of a single substring #L#, which indicates a L-toned TBU which is both at the beginning and the end of the string. The Chuave pattern L_{Ch} is thus captured by the following melody-local grammar.

(23) $G_{\mathrm{Ch}} = (S_{\mathrm{Ch}} = \{\}, M_{\mathrm{Ch}} = \{\#\mathrm{L}\#\})$

Any string that contains an H-toned TBU will not contain #L# in its melody. For example, for the well-formed string LLH, $\mathtt{mldy}(\mathrm{LLH}) = \mathrm{LH}$, which does not contain #L#. However, for the ill-formed string *LLLL, $\mathtt{mldy}(\mathrm{LLLL}) = \mathrm{L}$, which does contain #L#. Thus, M_{Ch} captures exactly the set of strings that conform to the Chuave pattern. Note that S_{Ch} is empty and thus always vacuously satisfied.

3.2 Culminativity in Arigibi and Prinmi

Tone patterns also exhibit interactions between local and long-distance constraints. First, Arigibi (Kiwai; Papua New Guinea Donohue, 1997) has a culminativity constraint: there can only be one H tone in a word. However, this H tone cannot spread. The relevant TBU here is the mora. Again, from the description in Donohue (1997) one can infer that the pattern holds on the surface.

(24) Arigibi (Donohue, 1997, p. 368)

a.	naː	‘finish’	e.	umú	‘dog’	h.	olaʔolá	‘red’
	L			LH			LLLH	
b.	tutuː	‘long’	f.	nímo	‘louse’	i.	tuniʔʌ́ʔʌ	‘all’
	LL			HL			LLHL	
c.	vovoʔo	‘bird’	g.	mudɛbɛ́	‘claw’	j.	idómai	‘eye’
	LLL			LLH			LHLL	
d.	ɛlaila	‘hot’	f.	ivío	‘sun’	k.	núʔʌtama	‘bark’
	LLLL			LHL			HLLL	
			g.	ŋgíʔɛpu	‘heart’			
				HLL				

The Arigibi pattern is distinguished from an obligatoriness constraint in that H-less words are allowed (such as (24d) [ɛlaila] ‘hot’). The set of well-formed TBU strings in Arigibi is given below in (25) as L_{Ar}.

(25) $L_{\text{Ar}} = \{\text{H, L, LL, LH, HL, LLL, LLH, LHL, HLL, LLLL, LLLH, LLHL, ...}\}$

Like UTP, this is non-local because we must ban any second occurence of a span of H tones. However, there is also a local constraint against H spans of two or more moras.

Thus, a melody-local grammar for L_{Ar} requires both a local constraint and a constraint on the melody. Like with UTP, we need to forbid HLH substrings in the melody, to prevent distinct H spans in the string. Thus, we posit $M_{\text{Ar}} = \{\text{HLH}\}$, identical to M_{UTP}. However, unlike UTP, we have to restrict H spans to only consist of a single TBU. For this, we posit a local constraint $S_{\text{Ar}} = \{\text{HH}\}$. This results in the grammar in (26).

(26) $G_{\text{Ar}} = (S_{\text{Ar}} = \{\text{HH}\}, M_{\text{Ar}} = \{\text{HLH}\})$

Examples of the interaction of these two constraints are given in Table 1. Recall that for a string to be well-formed with respect to G_{Ar}, it has to satisfy S_{Ar}, and the melody of that string has to obey M_{Ar}.

	w	$\texttt{mldy}(w)$	$S_{\text{Ar}} = \{\text{HH}\}$	$M_{\text{Ar}} = \{\text{HLH}\}$
a.	$\text{*L}\underline{\text{HH}}\text{HL}$	LHL	✗	✓
b.	*HLLHL	$\underline{\text{HLH}}\text{L}$	✓	✗
c.	$\text{*HL}\underline{\text{HH}}$	$\underline{\text{HLH}}$	✗	✗

Table 1 Ill-formed strings according to G_{Ar}. Offending substrings are underlined.

The string in Table 1a, *LHHHL, is ill-formed because it contains the substring HH and thus does not satisfy S_{Ar}. Thus, it is ill-formed regardless of the fact that $\texttt{mldy}(\text{LHHHL}) = \text{LHL}$ does satisfy M_{Ar}. In contrast, Table 1b, *HLLHL, satisfies S_{Ar} because it does not contain a HH sequence. However, $\texttt{mldy}(\text{HLLHL}) = \text{HLHL}$, which does not satisfy M_{Ar}, and so *HLLHL is ill-formed with respect to the grammar. It is also possible to violate both constraints, as in Table 1c, *HLHH.

Thus, G_{Ar} rejects any string with more than one span of Hs, and it rejects any string for which a span of Hs is more than one TBU long. However, any string with at most one H-toned mora will satisfy both S_{Ar} and M_{Ar}; thus, G_{Ar} describes exactly L_{Ar}. To illustrate, some examples from L_{Ar} are shown in Table 2.

	w	`mldy`(w)	$S_{\mathrm{Ar}} = \{\mathrm{HH}\}$	$M_{\mathrm{Ar}} = \{\mathrm{HLH}\}$
a.	LLL	L	✓	✓
b.	LLHL	LHL	✓	✓
c.	HLLLL	HL	✓	✓

Table 2 Well-formed strings according to G_{Ar}.

A similar constraint holds in Prinmi (Ding, 2006; Hyman, 2009). In Prinmi, words must have exactly one span of H tones.[8] This span can either be one or two moras long.

(27) Prinmi (Ding, 2006, p. 14)

a.	bɨ́bᴵobᴵo HLL	‘roasted flour with honey’	f.	bɨ́bᴵobᴵoge HLLL	‘as for roasted flour with honey’
b.	bɨ́ɬíɹu HHL	‘sunflower stem’	g.	bɨ́ɬípɜtsɨ HHLL	‘sunflower’
c.	tʃ’ɨn̥ɨ̃́dʒjɛ̃ LHL	‘dog-nose group’	h.	tʃ’ɨn̥ɨ̃́dʒjɛ̃ɹə LHLL	‘dog-nose ’ groups’
d.	tõpúk’ʉ́ LHH	‘donkey head’	m.	tõpúmɜ́ɬe LHHL	‘donkey tail’
e.	ɹətʃiʃṍ LLH	‘clean liquor’	n.	dʒjõdʒɨmɜ́ɬe LLHL	‘buffalo tail’

Ding (2006) and Hyman (2009) disagree with respect to the underlying representations in Prinmi, but they agree on the following generalization of the resulting surface pattern. All L-toned words are banned as are words with more than one H-tone span. Thus, H is culminative, as in Arigibi, but also obligatory, as in Chuave. Furthermore, whereas in Arigibi there is a constraint against H appearing on more than two moras, in Prinmi H can appear on either one or two moras. The set of possible well-formed words in Prinmi is thus L_{Pr} as shown in (28).

(28) $L_{\mathrm{Pr}} = \{$H, LH, HH, HL, LLH, LHL, LHH, HHL, HLL, LLLH, LLHL, LLHH, ...$\}$

Thus, like Arigibi and Chuave, L_{Pr} is long-distance in that there must be exactly one H-tone span in the word. Like Arigibi, it is also local in that there is a bound on the length of this H-tone span, except that in L_{Pr}, H-tone spans longer than two are banned.

The local constraint in Prinmi can be modeled with the local grammar S_{Pr} in (29).

(29) $S_{\mathrm{Pr}} = \{\mathrm{HHH}\}$

S_{Pr} contains the single forbidden substring HHH, which forbids a H-tone span from spreading more than two TBUs.

As H is obligatory in Prinmi, all-L words are also ill-formed. To capture this non-local constraint, the melody grammar for Prinmi must be $M_{\mathrm{Pr}} = \{\mathrm{HLH}, \#\mathrm{L}\#\}$, combining Arigibi’s constraint against two distinct H spans with Chuave’s constraint against L-toned melodies. Table 3 gives examples illustrating that strings that are ill-formed with respect to the Prinmi pattern are correctly rejected by a grammar G_{Pr} that combines S_{Pr} with this melody constraint. This G_{Pr} is given in (30).

(30) $G_{\mathrm{Pr}} = (S_{\mathrm{Pr}} = \{\mathrm{HHH}\}, M_{\mathrm{Pr}} = \{\mathrm{HLH}, \#\mathrm{L}\#\})$

	w	mldy(w)	$S_{\text{Pr}} = \{\text{HHH}\}$	$M_{\text{Pr}} = \{\text{HLH}, \#\text{L}\#\}$
a.	*L$\underline{\text{HHH}}$	LH	✗	✓
b.	*$\underline{\text{HHH}}$H	H	✗	✓
c.	*LLLL	$\underline{\text{L}}$	✓	✗
d.	*HHLLHH	$\underline{\text{HLH}}$	✓	✗
e.	*LHHLLLHH	L$\underline{\text{HLH}}$	✓	✗

Table 3 Ill-formed strings according to G_{Pr}. Offending substrings are underlined.

In Table 3a and 3b, *LHHH and *HHHH are correctly rejected by S_{Pr} because they contain the forbidden substring HHH, enforcing the constraint against a H-span that has spread to more than two TBUs.

The necessity of both a local and a melody constraint is shown in Table 3c through 3e. First, *LLLL does not run afoul of S_{Pr}, but as it contains only L-tones, its melody string is L, which contains the forbidden melody substring #L# and is thus ruled ungrammatical. The ill-formed string *HHLLHH satisfies S_{Pr}, as both H tone spans are exactly two TBUs long. However, it violates the melody grammar M_{Pr}, because mldy(HHLLHH) = HLH, which contains the forbidden melody substring HLH. The same goes for Table 3e. Table 4 illustrates how strings in L_{Pr} satisfy both S_{Pr} and M_{Pr}.

	w	mldy(w)	$S_{\text{Pr}} = \{\text{HHH}\}$	$M_{\text{Pr}} = \{\text{HLH}, \#\text{L}\#\}$
a.	HLLL	HL	✓	✓
b.	LLHH	LH	✓	✓
c.	LLLHHL	LHL	✓	✓

Table 4 Well-formed strings according to G_{Pr}.

In this way, the combined local and melody constraints of G_{Pr} work in concert to capture the generalization that words in Prinmi must have exactly one H-tone span, and that this span must be at most two TBUs long.

3.3 H-tone spread in Bemba

A striking case of a long-distance constraint interacting with bounded spreading occurs in Bemba (Bantu, Zambia; Bickmore and Kula, 2013, 2015). In phrase-final position in Bemba, the last H spreads unboundedly to the end of the word, while all preceding Hs undergo bounded spread (Bickmore and Kula, 2013, 2015). In Northern Bemba, in which the bounded spread is binary, this results in a surface pattern in which H spans are maximally of two TBUs in length, save for the rightmost, which necessarily extends to the end of the word. (In Copperbelt Bemba, the pattern is identical, except that bounded spread is ternary; Bickmore and Kula 2013, 2015. A similar pattern also occurs in Cilungu; Bickmore 2007, p. 8.)

The following data is from Northern Bemba, henceforth "Bemba." The TBU is the mora. First, a single H tone extends from its underlying position to the end of the word. The following data are listed as they are pronounced at the surface.

[8] In word-final position, this can technically be realized as a rising or falling tone; contours are abstracted away from here to focus on the long-distance nature of the pattern. For more on contours, see §6.2.

(31) Unbounded spreading in Bemba (Bickmore and Kula, 2013, pp. 105, 107)

a.	/tu-ka-pat-a/	'we will hate'	tu-ka-pat-a	LLLL
b.	/tu-lée-pat-a/	'we are hating'	tu-léé-pát-á	LHHHH
c.	/bá-ka-fik-a/	'they will arrive'	bá-ká-fík-á	HHHH

In terms of a surface generalization, the rightmost span of H-toned TBUs must extend to the end of the word. Thus, the surface forms of (31b) [tu-léé-pát-á] 'we are hating' and (31c) [bá-ká-fík-á] 'they will arrive' are LHHHH and HHHH, respectively; the underlying strings of TBUs, *LHLLL and *HLLL, are ill-formed.

That only the rightmost span of H-toned TBUs can extend more than two TBUs is shown in the following data, which each have two underlying H tones.[9]

(32) Bemba (Bickmore and Kula, 2013, p. 105,108)

a.	/béleng-á/	'read!'	béléeng-á	HHLH
b.	/tú-lub-ul-ul-é/	'we should explain '	tú-lúb-ul-ul-é	HHLLH
c.	/bá-a-pít-ile/	'they passed'	bá-a-pít-ílé	HLHHH
d.	twáá-ku-láa-pá	'we will be drawing (water)'	twáá-ku-láá-pá	HHLHHH

In all of the above examples, the rightmost span of Hs extends to the end of the word, whereas the preceding span extends to a maximum of two TBUs. Thus, for example, (32b) [tú-lúb-ul-ul-é] 'we should explain' is HHLLH, not *HHHLH. As (32d) [bá-a-pít-ílé] 'they passed' shows, bounded spreading respects the OCP, such that spans of H tones are separated by at least one L.

The crucial distinction here is between a string like *HHLLL and HHLLH (=32b): the former is ill-formed because the rightmost H-tone span does not extend to the right. This is a long-distance generalization, because a H-tone span must 'know' whether or not another H-tone span occurs to the right. However, it is also a local generalization, because all non-rightmost H-tone spans must extend to two TBUs (obeying the OCP); thus, *HLLLH and *HHHLH are also ill-formed.

Considering that all-L forms are also licit (as in (31a) [tu-ka-pat-a] 'we will hate' LLLL), the set of possible strings of TBUs that are well-formed according to the Bemba spreading generalization is as given as L_{Be} in (33).

(33) L_{Be} is exactly the set that is the union of

1. all strings of all Ls: {L, LL, LLL, LLLL ...}; and
2. all strings containing more than one H span, where the last H span continues to the end and any preceding H span is maximally of length two:
   { H, HH, HHH, ..., LH, LHH, LHHH, LLHH, LLLH, LHHHH, LLHHH, ...,
   HLH, HHLH, HLHH, HHLLH, HHLHH, LHHLH, HHLLHH, HHLLLH, ... }

This pattern is melody-local, as witnessed by the following grammar. First, bounded spreading is described by the local grammar in (34).

(34) $S_{\mathrm{Be}} = \{\#\mathrm{HLL}, \mathrm{LHLL}, \#\mathrm{HL}\#, \mathrm{LHL}\#, \mathrm{HHHL}\}$

The first two forbidden substrings in S_{Be}, #HLL and LHLL, capture the constraint that a H tone not already following a H tone cannot be followed by two L tones. This excludes, for example, strings like *HLLL and *LHLL. (Because of the OCP, it is possible that it is

[9] The second tone in (32a) and (32b) are 'melody high' tones assigned to a particular mora by tense, aspect, and mood morphology. As Bickmore and Kula (2013) explain, these tones behave identically to underlying tones with respect to the main spreading generalizations.

followed by a single L just in case another H tone follows, as in the string HLHH.) The next two forbidden substrings, #HL# and LHL#, capture the same constraint but at the end of the word; thus *LLHL is ill-formed but LLHH is not. Finally, S_{Be} also forbids the substring HHHL, which bans HHH sequences that are followed by an L. This following L is specified because HHH sequences *are* allowed in L_{Be}, just in case they are in a H span that reaches the end of the word.

As shown in Table 5, these local constraints force bounded spreading for any H span that does not reach the end of the word, but allow unbounded spreading if the H span does reach the end of the word.

	w	$S_{\mathrm{Be}} = \{\#\mathrm{HLL}, \mathrm{LHLL}, \#\mathrm{HL}\#, \mathrm{LHL}\#, \mathrm{HHHL}\}$
a.	*HLLH	✗
b.	*LHLLH	✗
c.	HHLLH	✓
d.	*HHHHLLL	✗
e.	LHHH	✓
f.	*HHLLL	✓

Table 5 Well-formed and ill-formed strings according to S_{Be}. For the ill-formed strings, offending substrings are underlined.

Table 5a and 5b contrast *HLLH and *LHLLH, in which a non-final H span is not binary, with HHLLH. The local grammar S_{Be} correctly marks *HLLH as illicit because it contains #HLL—i.e., the initial H has not spread—and *LHLLH because it contains the substring LHLL, indicating some word-medial H-span that is not binary. The well-formed HHLLH, in contrast, contains neither of these substrings. In Table 5d shows a string *HHHHLLL in which a non-final H has spread more than two TBUs. This is illicit because it contains the substring HHHL. This is contrasted with Table 5e, LHHH, which has a H that has spread more than two TBUs but spreads to the end of the word. This does not contain any of the substrings in S_{Be}, and is thus correctly judged well-formed.

Table 5f, however, is *in*correctly judged well-formed by S_{Be}: it has a non-final binary span of Hs, and so does not contain any of the substrings in S_{Be}. Herein lies the long-distance nature of the pattern: the local grammar S_{Be} cannot tell whether or not a H span is the last in the word. In other words, it cannot distinguish between HHLLH and *HHLLL. Thus, it cannot force unbounded spreading in the latter.

The melody constraint in M_{Be} in (35), however, can force unbounded spreading of the final H in the word.

(35) $M_{\mathrm{Be}} = \{\mathrm{HL}\#\}$

The forbidden melody substring HL# forbids a span of Ls intervening between a H and the end of the word. In other words, any final H span must spread to the end of the word. This is illustrated below in Table 6.

Table 6a and 6b show that the strings HHLLH and LHHH are well-formed with respect to M_{Be} because their melodies, HLH and LH, respectively, do not end in a HL sequence. In contrast, the melody of the ill-formed *HHLLL does: $\mathtt{mldy}(\mathrm{HHLLL}) = \mathrm{HL}$. Thus, the melody constraint M_{Be} forces spreading of the last H to the end of the word. This captures Bickmore and Kula (2015)'s generalization that a phrase-final TBU wants be associated to the final H tone in a word, if one exists.

	w	$\texttt{mldy}(w)$	$M_{\text{Be}} = \{\text{HL\#}\}$
a.	HHLLH	HLH	✓
b.	LHHH	LH	✓
c.	*HHLLL	$\underline{\text{HL}}$	✗

Table 6 Well-formed and ill-formed strings according to M_{Be}. For the ill-formed strings, offending substrings are underlined.

A combined table in Table 7 shows how a grammar G_{Be} (itself given in (36)) correctly distinguishes strings in L_{Be} from strings not in L_{Be}. Note that the well-formed strings (Table 7a through 7c) satisfy both S_{Be} and M_{Be}, while the ill-formed strings (Table 7d through 7f) violate at least one.

(36) $G_{\text{Be}} = (S_{\text{Be}} = \{\text{\#HLL}, \text{LHLL}, \text{\#HL\#}, \text{LHL\#}, \text{HHHL}\}, M_{\text{Be}} = \{\text{HL\#}\})$

	w	$\texttt{mldy}(w)$	$S_{\text{Be}} = \{\text{\#HLL}, \text{LHLL}, \text{\#HL\#}, \text{LHL\#}, \text{HHHL}\}$	$M_{\text{Be}} = \{\text{HL\#}\}$
a.	HHLLH	HLH	✓	✓
b.	LHHH	LH	✓	✓
c.	HHLHHH	HLH	✓	✓
d.	*$\underline{\text{LHLL}}$H	LHL	✗	✓
e.	*HHLLL	$\underline{\text{HL}}$	✓	✗
f.	*H$\underline{\text{HHHL}}$LL	$\underline{\text{HL}}$	✗	✗

Table 7 Well-formed and ill-formed strings according to G_{Be}. For the ill-formed strings, offending substrings are underlined.

Thus, through a combination of forbidden substring constraints operating both locally and over the melody, melody-local grammars can capture the blocking of unbounded H-tone spread in Bemba.

3.4 Karanga Shona non-assertive verb stems

Finally, one additional pattern that mixes a long-distance dependency with a local one is tone assignment in Karanga Shona (Bantu, Zimbabwe) verb stems. This pattern is interesting because it is not entirely melody-local. However, we can posit a melody-local approximation of the pattern that differs from the pattern in only a single form.

The pattern in question is for the non-assertive verb stems, which Odden (1984) states "is encountered in numerous morpho-syntactic environments" (p. 259). As in other Bantu languages, the stem in Shona verbs consists of the root plus extensions and the final vowel (Myers, 1987, p. 27). Stems can take on one of two patterns; this is usually analyzed as the result of the root being underlyingly H-toned or L-toned (the latter are sometimes analyzed as toneless, as in Hewitt and Prince 1989). Examples are given below with the prefixal complex /handáka/ 'I didn't'; the verb stem in each case is the information following the morpheme boundary.

(37) Karanga Shona non-assertive verb stems (Odden, 1984, p. 258)
H-toned root:

a.	handáka-pá	'I didn't give'
b.	handáka-tóra	'I didn't take'
c.	handáka-tóresá	'I didn't make take'
d.	handáka-tóréserá	'I didn't make take for'
e.	handáka-tóréséraná	'I didn't make take for each other'
f.	handáka-tóréséresaná	'I didn't make take a lot for each other'
g.	handáka-tóréséresesaná	(same as f)

L-toned root:

h.	handáka-biká	'I didn't cook'
i.	handáka-bikísa	'I didn't make cook'
j.	handáka-bikísíra	'I didn't make cook for'
k.	handáka-bikísísira	'I didn't make cook for each other'
l.	handáka-bikísísirana	'I didn't make cook a lot for each other'
m.	handáka-bikísírisisana	(same as l)

The two surface patterns for Karanga Shona verb stems are schematized as strings of H- and L-toned TBUs in (38).

(38) Karanga non-assertive verb stems (schematic)

H-toned	L-toned
H	
HL	LH
HLH	LHL
HHLH	LHHL
HHHLH	LHHLL
HHHLLH	LHHLLL
HHHLLLH	LHHLLLL
...	...

There are multiple derivational interpretations of this pattern. Odden (1984) analyzes it as the realization of a HHLB melody, where B is a variable that takes on the value of the root tone, and Hewitt and Prince (1989) analyze it as the 'edge-in' (Yip, 1988) mapping of one or two H tones, depending on the whether or not the root carries a tone, followed by the application of two bounded spreading rules to the initial H.

However, what is uncontroversial is the resulting surface pattern. Non-assertive verb stems either (39a) begin with a span of (maximally) three H-toned TBUs, followed by some number of L-toned TBUs, followed by a final H-toned TBU; or (39b) begin with (maximally) a LHH sequence of TBUs followed by some number of L-toned TBUs.

(39) a. H-toned roots: $\{\mathrm{H}, \mathrm{HL}, \mathrm{HLH}, \mathrm{HHLH}\} \cup \mathrm{HHHL}^m\mathrm{H}$, $m > 0$
b. L-toned roots: $\{\mathrm{L}, \mathrm{LH}, \mathrm{LHL}\} \cup \mathrm{LHHL}^n$, $n > 0$

The pattern as a whole is thus the union of the two sets in (39). This pattern must be identified by a learner in order to recognize that these verb stems belong to the same non-assertive category.

(40) $$L_{\mathrm{KS}} = \{\mathrm{H}, \mathrm{HL}, \mathrm{HLH}, \mathrm{HHLH}\} \cup \mathrm{HHHL}^m\mathrm{H} \cup \{\mathrm{L}, \mathrm{LH}, \mathrm{LHL}\} \cup \mathrm{LHHL}^n;\;\; m,n > 0$$

An important generalization here is that if a verb stem ends in a LH sequence, it cannot start with a LHH sequence (41a). Likewise, if a verb stem begins with a HHH sequence, it cannot end with a L (41b).

(41) a. *LHHLH b. *HHHL
*LHHLLH *HHHLL
*LHHLLLH *HHHLLL
... ...

This generalization is long-distance: whether or not the final TBU is H depends on whether or not the initial TBU is H, no matter how many L tones intervene. However, there is also a local component, in that the first span of H-toned TBUs can only extend to the third TBU.

The following shows that the pattern in the Karanga Shona non-assertive tense is largely melody-local, with the sole exception of two-TBU HL stems. This exception is discussed in more detail below. The following thus describes a set L'_{KS} that is identical to L_{KS} with the exception of the single string HL.

First, the constraints on the extent of the two H-toned spans is captured by the following local grammar S_{KS}.[10]

(42) $S_{\text{KS}} = \{\text{\#LL}, \text{\#HLL}, \text{\#HHLL}, \text{HHHH}, \text{LHLL}, \text{LHHH}, \text{HH\#}\}$

First, the forbidden substring #LL ensures that at least one of the first two TBUs is H, as shown below by the contrast between Table 8a and the grammatical forms in Table 8d and h. The forbidden substrings #HLL and #HHLL then ensure that, if the first TBU is H, then the second and third TBUs must be H as well (given a long enough string). This is illustrated by the contrast between the ungrammatical forms in Table 8b and c and the grammatical form in Table 8d below. Furthermore, the forbidden substring HHHH bans stretches of more than three H-toned TBUs, as illustrated in Table 8e.

	w	$S_{\text{KS}} = \{\text{\#LL}, \text{\#HLL}, \text{\#HHLL}, \text{HHHH}, \text{LHLL}, \text{LHHH}, \text{HH\#}\}$
a.	*$\underline{\text{LL}}$LLLH	✗
b.	*$\underline{\text{HLL}}$LLH	✗
c.	*$\underline{\text{HHLL}}$LH	✗
d.	HHHLLH	✓
e.	*$\underline{\text{HHHH}}$LH	✗
f.	*$\underline{\text{LHLL}}$LL	✗
g.	*$\underline{\text{LHHH}}$LL	✗
h.	LHHLLL	✓
i.	*H$\underline{\text{HH}}$	✗
j.	*HHHL$\underline{\text{HH}}$	✗

Table 8 Well-formed and ill-formed strings according to S_{KS}. For the ill-formed strings, offending substrings are underlined.

For strings that start with LH, the forbidden substrings LHLL and LHHH force the H-tone span to be exactly two TBUs long (in words of four TBUs or more), as illustrated in Table 8f, g, and h. Finally, the forbidden substring HH# bans a final stretch of H tones more than one TBU long, as shown in Table 8i and 8j.

We can then turn to the long-distance generalizations as outlined in (41). In terms of melodies, this means restricting well-formed melodies in longer forms to HLH and LHL. The melody grammar in (43) accomplishes this.

[10] The boundaries here are not strictly word boundaries; to make this explicit, one could replace these with appropriate boundaries that demarcate the stem.

(43) $M_{\text{KS}} = \{\text{\#HL\#}, \text{HLHL}, \text{LHLH}\}$

The first forbidden melody substring in (43), #HL#, requires that if a string starts with a H tone, the following span of L tones cannot be final. As shown in Table 9c and f below, this bans strings of the form *HHHL^n (recall from (41b)). The second constraint, HLHL, then ensures that words starting with a H tone have an HLH melody, as seen in the contrast between the well-formed string HHHLLH in Table 9d and the ill-formed string *HHHLLHL in Table 9d.

	w	$\texttt{mldy}(w)$	$M_{\text{KS}} = \{\text{\#HL\#}, \text{HLHL}, \text{LHLH}\}$
a.	HHHLLH	HLH	✓
b.	*HHHLL	<u>HL</u>	✗
c.	*HHHLLL	<u>HL</u>	✗
d.	*HHHLLHL	<u>HLHL</u>	✗
e.	LHHLLL	LHL	✓
f.	*LHHLLH	<u>LHLH</u>	✗
g.	*LHHLLLH	<u>LHLH</u>	✗

Table 9 Well-formed and ill-formed strings according to M_{KS}. For the ill-formed strings, offending substrings are underlined.

The final forbidden melody substring, LHLH, ensures that if a word starts with a LH sequence, it cannot end on a H. As shown in Table 9f and 9g, this bans strings of the form *LHHL^{n}H (recall from (41a)).

The melody-local grammar G_{KS} in (44) thus describes the set of strings L'_{KS} in (45).

(44) $$G_{\text{KS}} = (\, S_{\text{KS}} = \{\text{\#LL}, \text{\#HLL}, \text{\#HHLL}, \text{HHHH}, \text{LHLL}, \text{LHHH}, \text{HH\#}\},\; M_{\text{KS}} = \{\text{\#HL\#}, \text{HLHL}, \text{LHLH}\} \,)$$

(45) $$L'_{\text{KS}} = \{\text{H}, \text{HLH}, \text{HHLH}\} \cup \text{HHHL}^m\text{H} \cup \{\text{L}, \text{LH}, \text{LHL}\} \cup \text{LHHL}^n;\;\; m, n > 0$$

The pattern L'_{KS} is identical to L_{KS} with the exception of a single string, HL. This is banned by the melody constraint #HL#, but it is attested in Karanga Shona, as in (37b) [handáka-<u>tóra</u>] 'I didn't take' (stem with attested pattern underlined).

The literature on Shona does not describe any reason to distinguish these HL forms morphologically from other forms, nor independent evidence to treat them as exceptional. Thus, they are true cases of undergeneration for the melody-local grammar. There are a few ways to interpret this. As L_{KS} are L'_{KS} are identical for words of length three and more, the Karanga Shona pattern is melody-local in a general sense—and thus the long-distance aspects of the pattern can be learned with a melody-local learner.

More concretely, to deal with the local exception of HL words, we can modify satisfaction of a melody-local grammar to only check satisfaction of the melody grammar for words longer than the window examined by the local grammar—in this case, of length 4. That is, a string like HL, which is exactly of length 4 when the boundaries added, would not be evaluated for satisfaction of the melody. With this modification, L_{KS} would be described by G_{KS} exactly. How this modification could be extended to learning is discussed in §4.3.

Alternatively, it may be possible to deal with the single exception of HL forms in Karanga Shona with weighted constraints (as are briefly discussed in §5.3). Regardless, while Karanga Shona is not melody-local, a nearly identical pattern is.

3.5 Unattested long-distance patterns

Finally, while melody-local grammars can capture long-distance generalizations, they can only capture those which are definable with local constraints over melodies. The melody-local hypothesis thus makes a meaningful distinction between attested long-distance patterns and unattested, yet logically possible long-distance patterns.

One example of a logically possible, yet unattested pattern is the *midpoint pathology*, generable under some versions of ALIGNMENT constraints (Eisner, 1997; Buckley, 2009; Hyde, 2012). In the midpoint pathology, a H tone aligns itself to the center of a word.

(46) $L_{\text{MP}} = \{$LHL, LLHL, LHLL, LLHLL, LLLHLL, LLHLLL, LLLHLLL,...$\}$

In L_{MP}, words of odd length have a H-toned TBU in the center with equal number of L-toned TBUs on either side. Thus, for example, the seven-TBU word LLLHLLL is well-formed with respect to L_{MP}, but *LLHLLLL or *LLLLHLL is not. This is impossible to capture with a melody-local grammar, as it is impossible to regulate the number of L-toned TBUs on either side with local constraints. Thus, the melody-local hypothesis correctly excludes the midpoint pathology in its predicted typology.

Another example is first-last harmony (Lai, 2015). This is a logically possible pattern in which the first and last TBU in the word must agree in tone, regardless of the values of the intervening TBUs. This pattern is represented as the set of strings L_{FL} below in (47), which includes all strings that either both start and end with H, or start and end with L.

(47) $L_{\text{FL}} = \{$ H, HH, HLH, HHH, HLLH, HLHH, HHLH, HHHH, HLLLH, ...,
L, LL, LLL, LHL, LLLL, LLHL, LHLL, LHHL, LLLLL, ... $\}$

Such a pattern is not attested, at least in major surveys of the literature on tone (Yip, 2002; Hyman, 2011).[11] As Lai (2015) argues, the absence of first-last harmony in phonology is surprising, given the fact that word-initial and word-final position are perceptually prominent, and that many phonological generalizations refer to word edges or prominent positions (see, e.g., Beckman, 1998). This is no less the case for tone (de Lacy, 2002). In fact, recall that in the Karanga Shona pattern, a word that begins with a H tone must also end with a H tone. However, as shown in §3.4, this is the result of a fixed HLH melody, and thus does not allow arbitary intervening strings of tones.

The hypothesis that tone is melody-local thus explains the absence of first-last harmony. Banning all strings of the shape *H...L (and *L...H) is a long-distance constraint. However, this constraint cannot be expressed with forbidden substring constraints over a melody, because the melodies of these strings can be arbitrarily long as well: $\texttt{mldy}(\text{HLHL}) = \text{HLHL}$, $\texttt{mldy}(\text{HLHLHL}) = \text{HLHLHL}$, $\texttt{mldy}(\text{HLHLHLHL}) = \text{HLHLHLHL}$, ad infinitum. Thus, there is no finite set of forbidden substrings over the melody that can mark them all ill-formed.

3.6 Interim summary

Many tonal well-formedness patterns, including local and long-distance generalizations, share a property of melody-locality. That is, they can be described by the intersection of

[11] There are cases, e.g. in Cilungu (Bickmore, 2007, p. 16), in which a morpheme introduces two H tones, which associate to the beginning and end of the word. However, as both tones are introduced by a morphological process, this is best characterized as a kind of circumfixation and not phonological agreement. The applicability of melody-local learning to morphological processes is an interesting question for future work.

constraints holding either over an autosegmental melody or directly over the string of TBUs. Table 10 reviews the patterns surveyed in this and the preceding section and the melody-local grammars that describe them.

Pattern	Section(s)	Local grammar	Melody grammar
Kagoshima Japanese	§§2.1,2.2	{#L#, HH, HLL, HLH, LL#}	{}
UTP	§§2.3,2.4	{}	{HLH}
Chuave	§3.1	{}	{#L#}
Arigibi	§3.2	{HH}	{HLH}
Prinmi	§3.2	{HHH}	{HLH, #L#}
Bemba	§3.3	{#HLL, LHLL, #HL#, LHL#, HHHL}	{HL#}
Karanga Shona ($\geq 3\sigma$ words; L'_{KS})	§3.4	{#LL, #HLL, #HHLL, HHHH, LHLL, LHHH, HH#}	{#HL#, HLHL, LHLH}
Karanga Shona (all words; L_{KS})	§3.4	(not melody-local)	
First/last harmony	§3.5	(not melody-local)	
Midpoint pathology	§3.5	(not melody-local)	

Table 10 Summary of tone patterns surveyed in §§2 and 3. Patterns falling below the dotted line are unattested.

This section also showed how melody-locality excludes the unattested long-distance patterns H-tone centering and first-last harmony. Neither can be described through local constraints over a melody, and so the melody-local hypothesis explains their absence. As discussed in §3.4, Karanga Shona is an attested pattern that is not melody-local, although there is a melody-local pattern L'_{KS} that differs from the Karanga Shona pattern in only one string.

4 Learning melody-local grammars

This shared property of melody-locality makes the surveyed tone patterns efficiently learnable. This section demonstrates how this is possible. We first establish the criteria for learning, and then give an algorithm that can learn any melody-local pattern given a sufficient sample of positive data.

4.1 The learning framework

The learning criteria adopted here are those of Gold (1967)'s identification in the limit from positive data, which are roughly as follows. The Gold framework considers learners (learning algorithms) that take as input a finite set of examples of a pattern (e.g., a finite set of strings drawn from a target infinite set of strings) and generalize a grammar representing a (potentially infinite) pattern. A class of patterns is Gold-style learnable if there is some learner such that, for every pattern in the class, there is some finite sample for which the learner returns that pattern exactly. In other words, there is a learner that is guaranteed to learn every pattern in the class, given a sufficient sample of that pattern. Finally, a cognitively plausible learning algorithm computes in a reasonable amount of time. The algorithm must, then, also be efficient (de la Higuera, 2010). Briefly, this means that the number of steps the algorithm takes will be at most polynomially as large as the size of the data it is given.

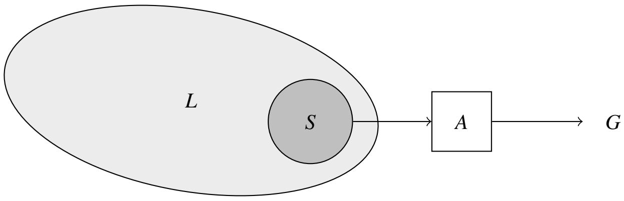


Fig. 1 Model of Gold-style learning. A learner A takes as input a finite sample S of pattern L and generalizes to a grammar G. The learner is successful if, given that S is a sufficient sample, G describes L.

Gold-style learning is an abstraction in some ways. First, the learner must match the target pattern exactly. However, a learner satisfying the Gold requirement is *guaranteed* to learn the pattern given any set of examples that contains a sufficient sample of data. Furthermore, this guarantee extends to any pattern in the class the learner is designed to learn. Thus, the learner's behavior is well-understood—we know how it will behave on a range of samples and a range of patterns.

Second, the learner is given a 'perfect' sample, in that all of its data is representative of the target pattern, and all aspects of the pattern are represented. Thus, a Gold-style learner says nothing about how to learn in the face of exceptions. This represents, however, a factorization of the learning problem: we understand first how to learn the class of patterns itself, and then we can use this knowledge to posit gradient or statistical learners that can learn these patterns in the face of exceptions. In other words, stochastic learners can use the structure of categorical learners. For example, Heinz (2010a) defines a categorical precedence-based learning procedure for long-distance phonotactics (to be discussed in more detail in Sec. 5), and Heinz and Rogers (2010) show how this can be extended to a stochastic learner that operates in the same way. Similarly, stochastic maximum entropy learners that consider learning constraints over tiers (Hayes and Wilson, 2008; Gallagher and Wilson, 2018) use the same fundamental structure as tier-based strictly local grammars (Heinz et al., 2011).

The following describes a learner for melody-local patterns that satisfies all of the criteria outlined above: given a finite, representative set of examples from a melody-local pattern, it converges to that exact pattern. It also does this in a linear number of steps with respect to the size of the data, and is thus very efficient.

4.2 Strictly local learning

The learner posited here is based entirely on the learning of strictly local patterns (García et al., 1990). This proceeds as follows. First, we fix some number k. Given a string, we can generate its substrings of length k using the same scanning procedure described in §2.2. That is, we take a window of length k and shift it from left to right through the word. The following shows that the 3-substrings (substrings for which $k = 3$) of #LLHL# are $\{$#LL, #LLH, LHL, HL#$\}$.

(48)

$\boxed{\text{\# L L}}$H L #	# $\boxed{\text{L L H}}$L #	# L$\boxed{\text{L H L}}$#	# L L$\boxed{\text{H L \#}}$
#LL	LLH	LHL	HL#

More examples are given in Table 11.

String	Substrings of length 3
#LLHL#	{#LL, LLH, LHL, HL#}
#LHL#	{#LH, LHL, HL#}
#LHLL#	{#LH, LHL, HLL, LL#}
#LLLH#	{#LL, LLL, LLH, LH#}
#LLLLH#	{#LL, LLL, LLH, LH#}

Table 11 Examples of 3-substrings for several strings.

This scanning procedure takes linear time in the size of the data—that is, the amount of time it takes to generate substrings from a set of data is directly proportional to the size of the data. This is extremely efficient.

Given k, the learning procedure is very simple. First, the learner sets its initial hypothesis for the grammar to be the set of all possible k-substrings. The size of this set will always be finite, given a finite set of symbols used to create the substrings. For $k = 3$ the initial hypothesis of the learner will be the set below in (49).

(49) Initial hypothesis for S when $k = 3$
{ #H#, #L#, #HH, #HL, #LL, #LH, HHH, HHL, HLH,
HLL, LHH, LHL, LLH, LLL, HH#, HL#, LH#, LL# }

This means that the learner begins by hypothesizing that *all* forms are ungrammatical. The learner then takes as input examples of the target pattern, keeping track of the k-substrings it observes in the input examples. These *observed k-substrings* are then removed from its hypothesis for the grammar. That is, it considers the observed k-substrings to now be licit. This process is shown for $k = 3$ and examples from the penultimate-or-final H pattern L_{KJ} from Kagoshima Japanese in Table 12.

Table 12 shows the status of the learner after being shown six input strings from the L_{KJ} pattern (in arbitrary order). At each step (i.e., after each new string from the sample), the learner updates its hypothesis by removing all of that string's substrings from its hypothesis. For example, in Step 2, after seeing the string #LLHL#, the learner now treats #LL, LLH, LHL, and HL# as licit substrings. Note that the learner *remembers* from step to step the substrings it has seen from previous steps, so the hypothesized grammar of forbidden strings either stays the same or grows smaller with each step. Because the learner only *subtracts* from the hypothesis for the grammar, and never adds to it, means that the learner is *monotonic* (Heinz, 2010b). This means that it only considers increasingly general patterns, and thus avoids the 'subset problem' (Angluin, 1980).

By Step 6, the learner has converged to a grammar that is equivalent to S_{KJ}. That is, it has *generalized* to a grammar that describes an infinite set of strings. For example, the resulting grammar in Table 12 accepts the string #LLLLLHL#, even though this was not given in the input data. This is true for any string in L_{KJ}; thus, the learner has generalized to an infinite pattern from a finite set of data.

Note that, because the learner is operating under the assumption that $k = 3$ that it has converged to a grammar that is intensionally distinct from the grammar S_{KJ} given in (9) in §2.2. Namely, whereas S_{KJ} included the 2-substring HH, the grammar in Table 12 instead forbids all 3-substrings that include HH: #HH, LHH, HHL, and HHH. However, again, these two grammars are extensionally equivalent, in that they describe exactly the same pattern.

Step	String	Hypothesis
0.		{#H#, #L#, #HH, #HL, #LL, #LH, HHH, HHL, HLH, HLL, LHH, LHL, LLH, LLL, HH#, HL#, LH#, LL#}
1.	#H#	{#H#, #L#, #HH, #HL, #LL, #LH, HHH, HHL, HLH, HLL, LHH, LHL, LLH, LLL, HH#, HL#, LH#, LL#}
2.	#LLHL#	{#H#, #L#, #HH, #HL, #LL, #LH, HHH, HHL, HLH, HLL, LHH, LHL, LLH, LLL, HH#, HL#, LH#, LL#}
3.	#LHL#	{#H#, #L#, #HH, #HL, #LL, #LH, HHH, HHL, HLH, HLL, LHH, LHL, LLH, LLL, HH#, HL#, LH#, LL#}
4.	#LLLH#	{#H#, #L#, #HH, #HL, #LL, #LH, HHH, HHL, HLH, HLL, LHH, LHL, LLH, LLL, HH#, HL#, LH#, LL#}
5.	#LLLLH#	{#H#, #L#, #HH, #HL, #LL, #LH, HHH, HHL, HLH, HLL, LHH, LHL, LLH, LLL, HH#, HL#, LH#, LL#}
6.	#HL#	{#H#, #L#, #HH, #HL, #LL, #LH, HHH, HHL, HLH, HLL, LHH, LHL, LLH, LLL, HH#, HL#, LH#, LL#}
		= {#L#, #HH, HHH, HHL, HLH, HLL, LHH, HH#, LL#}

Table 12 Strictly local learning from a sample of L_{KJ} and where $k = 3$. Grayed out text indicates forbidden substrings that have been removed from the hypothesis of the learner at each step.

For a target strictly local pattern describable with forbidden substrings of length k, this learning method is guaranteed to converge to a grammar representing exactly the target pattern, given examples that show all of the k-substrings in the pattern that are not forbidden (García et al., 1990). Furthermore, as already noted, it is extremely efficient. Importantly, the learner requires knowing k in advance. This is not an uncommon assumption for phonological learners (Hayes and Wilson, 2008). Also, it is a logical necessity: without knowing k in advance, strictly local learning is impossible (Rogers and Pullum, 2011; Rogers et al., 2013). However, we can posit that k for scanning and learning can be seen as coming from constraints on working memory, e.g. Miller (1956)'s 'magical number' of 7 ± 2 objects. All of the examples here conform to this generalization—a k of at most 4 suffices.

4.3 Melody-local learning

Having established a method for strictly local learning, we can extend it directly to learning constraints on the melody. To learn a melody-local grammar, which consists of both local constraints over a surface string and constraints over its melody, the learner learns both simultaneously. Learning the local constraints proceeds as outlined in the previous section. Learning the melody constraints proceeds in an almost identical fashion. We fix a length j for melody learning. This can be distinct from the k for local learning, but to simplify the exposition we shall assume $k = j$. The learner then has a hypothesis M for its melody grammar, and updates this by removing from M all j-substrings it sees *in the melody* of each string.Thus, the melody-local learner follows the procedure in (50) at each step.

(50) For each data point w,
1. update hypothesis S using the k-substrings of w, and
2. update hypothesis M using the j-substrings of $\texttt{mldy}(w)$.

Because we are assuming the OCP, the learner's initial hypothesis for the melody grammar need not contain adjacent sequences of Ls and Hs. Thus, for example, when $j = 3$ the initial state of the hypothesis for the melody grammar is as follows.

(51) Initial hypothesis for M when $j = 3$
$\{$#H#, #L#, #HL, #LH, LHL, HLH, HL#, LH#$\}$

To illustrate how this works, let us look at an example involving an interaction between the local constraints and the constraints on the melody. Recall that in Prinmi, a local constraint against a H span spreading more than two moras combined with a constraint that required exactly one H-tone span per word. Thus, strings like *LHHH are ill-formed, because a H-tone has spread more than two TBUs, and strings like *LLLL and *LHLLH are ill-formed, because they do not contain exactly one H-span. Recall this is a non-local constraint, because it bans *any* two H-toned TBUs separated by at least one L-toned TBU. Strings like the set in (52), however, are well-formed.

(52) LHHL, HLL, LLHLL, H, HH, LLLH

The learning procedure outlined in (50) learns a grammar equivalent to the grammar G_{Pr} in (30), given the strings in (52) as data points and $k = j = 3$. The steps for local learning are given in Table 13, and the steps for melody learning are given in Table 14.

Step	String	Hypothesis for local grammar
0.		{#H#, #L#, #HH, #HL, #LL, #LH, HHH, HHL, HLH, HLL, LHH, LHL, LLH, LLL, HH#, HL#, LH#, LL#}
1.	#LHHL#	{#H#, #L#, #HH, #HL, #LL, #LH, HHH, HHL, HLH, HLL, LHH, LHL, LLH, LLL, HH#, HL#, LH#, LL#}
2.	#HLL#	{#H#, #L#, #HH, #HL, #LL, #LH, HHH, HHL, HLH, HLL, LHH, LHL, LLH, LLL, HH#, HL#, LH#, LL#}
3.	#LLHLL#	{#H#, #L#, #HH, #HL, #LL, #LH, HHH, HHL, HLH, HLL, LHH, LHL, LLH, LLL, HH#, HL#, LH#, LL#}
4.	#H#	{#H#, #L#, #HH, #HL, #LL, #LH, HHH, HHL, HLH, HLL, LHH, LHL, LLH, LLL, HH#, HL#, LH#, LL#}
5.	#HH#	{#H#, #L#, #HH, #HL, #LL, #LH, HHH, HHL, HLH, HLL, LHH, LHL, LLH, LLL, HH#, HL#, LH#, LL#}
6.	#LLLH#	{#H#, #L#, #HH, #HL, #LL, #LH, HHH, HHL, HLH, HLL, LHH, LHL, LLH, LLL, HH#, HL#, LH#, LL#}
		= {#L#, HHH, HLH}

Table 13 Learning local grammar for L_{Pr}; $k = 3$.

As $k = 3$, the local portion of the learner begins with the hypothesis in (49); that is, the same as for learning L_{KJ}. As the data comes in, it removes the 3-substrings it sees in each data point from this hypothesis. By Step 6, it has converged to a grammar that only includes #L#, HHH, and HLH as forbidden substrings, which is correct for L_{Pr}. (Note that, while extensionally equivalent, this grammar is not identical to the grammar S_{Pr} given for Prinmi in (29) in §3.2. How to determine the identical grammar is given below.)

In terms of melody learning, for the same sample of data the learner will correctly converge to the melody grammar M_{Pr} for L_{Pr} originally given in (30). This is shown in Table 14.

Step	String	Melody	Hypothesis for melody grammar
0.			{#H#, #L#, #HL, #LH, LHL, HLH, HL#, LH#}
1.	#LHHL#	#LHL#	{#H#, #L#, #HL, #LH, LHL, HLH, HL#, LH#}
2.	#HLL#	#HL#	{#H#, #L#, #HL, #LH, LHL, HLH, HL#, LH#}
3.	#LLHLL#	#LHL#	{#H#, #L#, #HL, #LH, LHL, HLH, HL#, LH#}
4.	#H#	#H#	{#H#, #L#, #HL, #LH, LHL, HLH, HL#, LH#}
5.	#HH#	#H#	{#H#, #L#, #HL, #LH, LHL, HLH, HL#, LH#}
6.	#LLLH#	#LH#	{#H#, #L#, #HL, #LH, LHL, HLH, HL#, LH#}
			= {#L#, HLH}

Table 14 Strictly local melody learning from the sample of L_{Pr} from Table 13; $j = 3$.

As shown in Table 14, the melodies for the strings in (52) are LHL, HL, H, and LH, from which the learner will extract all possible melody 3-substrings except for #L# and HLH. Its hypothesis for the melody grammar thus converges to $\{\#\mathrm{L}\#, \mathrm{HLH}\}$, which identical to M_{Pr}. Note that the learner is guaranteed to not change this hypothesis. For any example string from the pattern L_{Pr}, its melody will never contain the 3-substrings #L# and HLH. Thus, the learner will never remove these 3-substrings from its hypothesis.

Thus, the learning procedure outlined in (50) returns the following grammar.

(53) $G = (\ S = \{\#\mathrm{L}\#, \mathrm{HLH}, \mathrm{HHH}\},\ \ M = \{\#\mathrm{L}\#, \mathrm{HLH}\}\ )$

This grammar is equivalent to the grammar G_{Pr} in (30) for L_{Pr}—all and only the strings in L_{Pr} satisfy the grammar G in (53). Again, this is not identical to G_{Pr}: S includes HLH and #L#, which are the result of the culminativitiy and obligatoriness constraints on H. To obtain G_{Pr} exactly, we can simply apply a minimization procedure in which we remove from S any substring that also appears in M. Because any such forbidden melody substring in M will also forbid the same local sequence of TBUs as S, this will not change the pattern described by the grammar. This minimization procedure will obtain S_{Pr} exactly.

We have thus seen how the learning procedure in (50) correctly and efficiently learns a melody-local pattern given finite samples of data. While this and L_{KJ} in the previous section were only two examples, we can *guarantee* that this learning will always happen, given a sufficient sample of data. A full proof is given in the Appendix, but the basic idea has already been illustrated in the above examples. If the learner sees all local k-substrings and all j-substrings in the melody that are *not* forbidden in the pattern, then it will correctly remove them from its local and melody hypotheses for the grammar. Conversely, given data points from the sample pattern, it will never see any k- or j-substrings that are forbidden locally or in the melody, respectively. It will thus correctly never remove these k- and j-substrings from its local and melody hypotheses for the grammar. This guarantees that, given samples exhibiting all allowed k- and j-substrings in the pattern, it will always converge to the correct grammar.

Finally, how would the learner behave when presented with the data from Karanga Shona non-assertive pattern, which was shown in §3.4 not to be entirely melody-local? Recall that the issue with the Karanga Shona pattern was that, in general, words of the form $\mathrm{H}^n\mathrm{L}^m$ are forbidden, with the single exception of the word HL. Thus, the grammar G_{KS} that approximated Karanga Shona included #HL# as a forbidden sequence its melody grammar M_{KS}. However, given data from Shona, the learning procedure outlined above would overgeneralize from a single example HL. Assuming $k = j = 4$, the learner would, upon seeing an input HL, remove #HL# from its hypothesis for the melody grammar. This would thus return a grammar that incorrectly accepts words of the form HHHL^n.

One solution to this problem, based on the discussion in §3.4, is to have the melody portion of learning only apply to strings of greater than length k (when the boundaries are included). As discussed in §3.4, the melody grammar is only necessary for capturing long-distance dependencies. Thus, we can posit a variation of the learner that only modifies its hypothesis for the melody grammar based on strings greater than length k (that is, the size the local grammar is sensitive to). This would make an input HL string, which is of length 4 when the boundaries are added, 'invisible' to the melody learning procedure. Thus #HL# would never be removed from the hypothesis for the melody grammar, and the learner would return a grammar extensionally equivalent to that of G_{KS}.

5 Comparison to other approaches

Previous approaches to long-distance learning in phonology have appealed to either precedence relations (Heinz, 2010a), tier projection (Hayes and Wilson, 2008; Goldsmith and Riggle, 2012; Jardine and Heinz, 2016; McMullin and Hansson, 2015; Jardine and McMullin, 2017), or some combination of both (Graf, 2017). However, neither can learn the full range of tonal patterns discussed in this paper. The conclusion is thus that, moving forward, either categorical or statistical approaches to learning tone must somehow incorporate the melody-local structure proposed in this paper.

5.1 Tier-projection and precedence

We focus mainly on tier projection. Tier projection implements a notion of relative locality by ignoring some subset of units in the surface string (Heinz et al., 2011). For example, we can capture L_{Ar} (the Arigibi pattern in which two H-toned TBUs are banned) using a tier projection in which L-toned TBUs are ignored. This is depicted in (54).

(54) a.
$$\begin{array}{ccccc} & \mathrm{H} & & & \\ & \uparrow & & & \\ \mathrm{L} & \mathrm{H} & \mathrm{L} & \mathrm{L} & \mathrm{L} \end{array}$$
b. *
$$\begin{array}{ccccc} & \mathrm{H} & & & \mathrm{H} \\ & \uparrow & & & \uparrow \\ \mathrm{L} & \mathrm{H} & \mathrm{L} & \mathrm{L} & \mathrm{H} \end{array}$$

In (54a), the tier projected for the string LHLLL is H, whereas for *LHLLH (which is ill-formed according to L_{Ar}) it is HH. We can thus describe all and only the strings in L_{Ar} by a forbidden substring grammar $\{\mathrm{HH}\}$ that is interpreted as holding over the tier projection that only includes H-toned TBUs. In terms of formal language theory, such patterns are called tier-based strictly local (Heinz et al., 2011), and are efficiently learnable even if the tier is not specified in advance (Jardine and Heinz, 2016; Jardine and McMullin, 2017). Other models of long-distance phonotactic learning use this basic idea of tier projection, even though it is usually augmented with feature representations and statistical learning methods (Hayes and Wilson, 2008; Goldsmith and Riggle, 2012; Gallagher and Wilson, 2018).

However, it can easily be shown that tier projection is not sufficient for the other long-distance tone patterns discussed in this paper. First, consider the same tier projection for unbounded tone plateauing pattern, L_{UTP}, in which every string must have at most one span of H-toned TBUs. The examples below in (55) show the tier projection for a string that is well-formed with respect to L_{UTP} (55a) and that for a string that is ill-formed (55b).

(55) a. H H H b. * H H H
↑ ↑ ↑ ↑ ↑ ↑
L H H H L L H H L L L H

As shown in (55a) and (55b), a string LHHHLL with a single, unbroken stretch of three H-toned TBUs projects a tier HHH, as does an ill-formed string *HHLLLH with one span of two H-toned TBUs followed by another, distinct span of a single H-toned TBU. Thus, tier projection cannot distinguish between a single plateau and multiple, distinct plateaus, because the intervening L-toned TBUs (or lack thereof) have been ignored.

This same issue occurs in Prinmi, in which words can contain only at most one span of H-tones. As Prinmi allows spreading of H-tone up to two TBUs, it is possible for there to be more than one H-toned TBU to appear in the string, as shown below in (56a). Thus, we cannot ban HH sequences on a projection of H-toned TBUs.

(56) a. H H b. * H H
↑ ↑ ↑ ↑
L H H L L L H L L H

However, as shown in (56b), ill-formed strings like *LHLLH do contain two Hs on the tier projection. Thus, a grammar operating over this tier projection cannot distinguish between (56a) and (56b). As in L_{UTP}, ignoring the L-toned TBUs loses the local information that distinguishes single spans of TBUs from distinct spans of single H TBUs.

This is also the case for L_{Be}, the long-distance constraint blocking unbounded spread in Bemba. The examples in (57) compare the tier projections for strings that are well-formed (57a, b) and ill-formed (57c) with respect to this pattern.

(57) a. H H H b. H H c. * H H
↑ ↑ ↑ ↑ ↑ ↑ ↑
H H L L L H L L L H H H H L L L L

Finally, although the Karanga Shona pattern L_{KS} was not melody local, it is not capturable by tier projection either. For example, recall that in L_{KS} strings of the form LHHL^n are well-formed but strings of the form *LHHL^{n}H are not. We can forbid the latter with a constraint *HHH on a H-TBU tier projection, as shown in (58) below.

(58) a. H H b. * H H H
↑ ↑ ↑ ↑ ↑
L H H L L L L H H L L H

However, this both allows ill-formed strings of, for example, the form *L^{n}HH, and incorrectly bans well-formed strings of the form HHHL^{n}H, as shown in (59) below.

(59) a. * H H b. H H H H
↑ ↑ ↑ ↑ ↑ ↑
L L L L H H H H H L L H

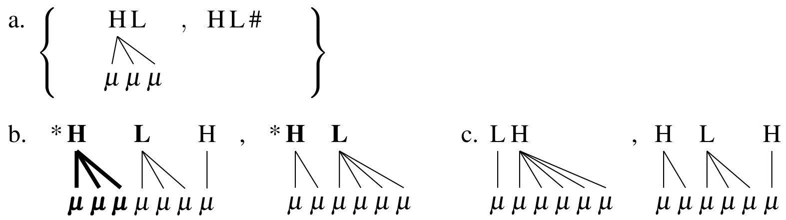
Again, this is due to the fact that tier projections ignore local information: a tier including only H-toned TBUs fails to distinguish between well-formed H-toned spans at the beginning of the word versus ill-formed H-toned spans at the end of the word. Thus, the Karanga Shona pattern is not capturable by tier-projection, and neither is its melody-local approximation, L'_{KJ}.

To summarize, several of the tone patterns surveyed here are impossible to characterize with grammars that operate over tier projections. This is because these patterns have both a local and long-distance component, and tier projections cannot capture the local components. It bears emphasizing that tier projections are not entirely absent from the representation from tone. The strings of TBUs used throughout this paper are themselves a kind of tier projection, as they abstract away from segmental information in the string. However, the above examples show that tier projections are not *sufficient* for representing tone.

Another proposal for learning long-distance generalizations is precedence-based learning, a la Heinz (2010a). Briefly, a precedence-based grammar forbids sub*sequences* of the form *x*...*y*, which is interpreted as banning *x* preceding *y* anywhere in the string, ignoring any intervening material. For example, L_{UTP} is describable by a precedence grammar forbidding the subsequence H...L...H (Graf, 2017). Heinz (2010a) and Heinz and Rogers (2013) show how these grammars are efficiently learnable in a similar way to strictly local grammars. However, obligatoriness constraints like Chuave cannot be captured by precedence grammars (Rogers et al., 2013). Furthermore, patterns like Prinmi and Bemba, which also include local generalizations, cannot be captured by precedence grammars, as precedence grammars are necessarily 'blind' to local information (and thus neither can they be captured by Graf (2017)'s interval-based strictly piecewise grammars, which generalize precedence and tier projection grammars).

5.2 Learning over autosegmental representations

Finally, Jardine (2017) presents a class of local AR grammars which can capture a range of local and long-distance constraints and their interactions (Jardine, 2019). These grammars are based on forbidden sub*graphs* of ARs, as ARs are graphs (Coleman and Local, 1991). A subgraph is simply a connected piece of an AR. For example, if we view the Bemba pattern in terms of ARs, we can model the constraints against bounded and unbounded spreading with the forbidden AR subgraphs in (60a).[12]

(60)


The structures in (60a) specify a nonfinal H tone that has spread more than two TBUs and a melody in which ends in a HL sequence, respectively. As highlighted in bold in (60b), ARs in which a nonfinal H tone has spread more than two TBUs, or in which a final H tone has not spread to the end of the word, contain one of these as subgraphs. Thus, the grammar in (60a) marks them as ill-formed. As shown in (60c), ARs that are well-formed with respect to the pattern do not contain either subgraph.

Similarly, ARs can also capture the case in Karanga Shona that was problematic for melody-local grammars (§3.4). Recall that in Karanga Shona, a HL melody was allowed just in case the H does not spread: HL is a well-formed string, but strings of the form *HHHL^n are not. As discussed in §3.4, a #HL# melody constraint is needed to ban the latter, but also

[12] For brevity, this grammar abstracts away from the complete set of constraints that obtain bounded spreading.

excludes the former. A grammar based on banned AR subgraphs, however, can capture this behavior by making this melody constraint more specific:

(61) # H L #

$\mu\ \mu$

The constraint in (61) bans multiple association of a H tone when it is part of a HL melody. This would forbid *HHHL^n forms while allowing HL.[13]

Thus, forbidden subgraph grammars over AR properly include the melody-local property: they can capture all of the patterns discussed in this paper. However, learning with graph grammars is still a poorly understood topic (Eyraud et al., 2012). In the general case, searching for subgraphs can be computationally taxing. Thus, while ARs are highly structured (Kornai, 1995; Jardine and Heinz, 2015), they are still more complex than strings, and so there is no guarantee that there is an efficient way to search them for observed subgraphs.

Finally, as to be discussed in the following section, forbidden subgraph constraints over ARs appear to be *too* expressive—that is, they can describe unattested well-formedness patterns that melody-local grammars cannot. Future work can explore restrictions on AR grammars that can make them efficiently learnable and restrictive. However, any such restriction must be at least able to capture melody-local patterns, in order to capture the tone patterns discussed here.

5.3 Implications for phonological theory

Previous sections have demonstrated that melody-locality is a necessary property for learning tone, and this section has shown that tier projection learners do not have this property. Tier projection learners, whether they are categorical (as in Jardine and Heinz 2016; Jardine and McMullin 2017) or probabilistic (as in Hayes and Wilson 2008; Goldsmith and Riggle 2012; Gallagher and Wilson 2018; Gouskova and Gallagher pear), are guaranteed to fail to learn the patterns discussed in this paper, because these patterns simply cannot be represented by tier projections. The same goes for precedence-based learning (Heinz, 2010a; Heinz and Rogers, 2010) or interval-based learning (Graf, 2017).

Thus, future models of learning must incorporate the property of melody-locality if they are to learn tone patterns like the ones surveyed in this paper. Learning directly over ARs incorporates the melody-local property, but there are unsolved problems with efficiency and overgeneration.

A consequence of the results in this paper is thus that future work on maximum entropy models of tone should incorporate a melody-local structure. This will be straightforward. The analysis of Wargamay in Hayes and Wilson (2008), for example, uses constraints that operate over distinct projections (Table 18, p. 419). We can thus incorporate melody constraints by considering two different ‘projections,’ in the terms of Hayes and Wilson: one which operates over the string of TBUs and the other which operates over the string generated by the `mldy` function. As an example, a schematic for a maximum entropy analysis of Prinmi is given in Table 15.

One constraint is *HHH, which would be evaluated over each candidate string w, and one constraint is *HLH, which would be evaluated over $\mathtt{mldy}(w)$. Each can be given their

[13] A full analysis would also require AR versions of the local spreading constraints in (42), to eliminate forms of the shape *HLL^n. However, this is still describable with AR subgraph grammars; see Jardine (2017).

	Constraint	Projection	Comment
1	*HHH	surface string	H-spans cannot be more than three TBUs long
2	*HLH	melody	No two H-spans in a string

Table 15 Schematic for MaxEnt constraints for Prinmi

own weight, and then violations for these two constraints can be summed in the usual way. A maximum entropy learner could then learn these weights independently, thus implementing in a stochastic framework the core melody-local property of melody constraints operating independently of local constraints. How melody-local learners behave under gradient constraint learning is thus an immediately approachable avenue for future research made available by the current results. One specific open question is whether a melody-local learner with weighted constraints can learn the full Karanga Shona pattern.

6 Discussion

This paper has represented tone patterns in terms of strings, which raises several representational questions. This section addresses several of these questions.

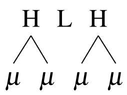
6.1 Autosegmental representations and strings

In melody-local grammars, local constraints are represented in terms of strings of H- and L-toned TBUs. This may seem counter to the long-standing idea that tone should be represented in terms of ARs (though see Cassimjee and Kisseberth 1998; Shih and Inkelas ming). However, this is not the case: we can view these string representations as a 'shorthand' for ARs. While human learners may indeed be using AR structures to learn, we can at least approximate how they are learning with string grammars.

In fact, there is reason to believe that melody-local grammars more accurately capture the independent behavior of melody constraints and local constraints. As can be seen in the forbidden subgraph grammar given in (60a) in the previous section, ARs can refer to both melody information and local associations to TBUs. Thus, these constraints can express arbitrary combinations of non-local and local information. This appears to be too expressive for tonal phonotactics.

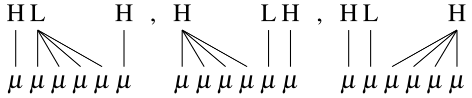
For example, consider the hypothetical forbidden AR subgraph constraint below.

(62)



The constraint in (62) enforces a 'no consecutive spreading Hs' pattern: if a H has spread, then the preceding or following H cannot also spread. To illustrate, (63) contrasts ARs that are well-formed with respect to this constraint (631) with an AR that is ill-formed (632).

(63) 1.



2. *H L H

μ μ μ μ μ μ μ

As shown in (631), given two consecutive H-tone spans, this constraint is satisfied when *at most* one span is more than one TBU long. It is violated, then, when both consecutive H-tone spans are more than one TBU long.

This constraint thus implements a bidirectional “if-then” pattern: if a H tone spreads, then neither the preceding nor the following can spread. Spreading in tone does not behave in this way; major reviews of the typology of tonal patterns (Yip, 2002; Hyman, 2011), for example, do not list any spreading patterns that follow such a rule. Thus, forbidden subgraph grammars over ARs overgenerate an unattested pattern, through constraints that can simultaneously refer to non-local melody information as well as local associations between tones and TBUs.

In contrast, this pattern is not melody-local. This is because the constraint in (62) simultaneously refers to both local and melody information: a HLH melody in which both H tones have spread. Because melody-local grammars independently represent constraints on spreading and constraints on the melody, it is impossible to forbid exactly this configuration. (For a proof that this is not melody-local, see the Appendix.)

Thus, melody-local grammars are more restrictive than local grammars over ARs because they constrain the interaction between long-distance and local constraints. This restriction appears to be the correct one. Again, this is not to say that humans do *not* use ARs to represent tonal information in any aspect of the phonological grammar. However, the melody-local model compares favorably in terms of learning because it is more restrictive and there is a known method for efficiently learning melody-local grammars.

6.2 Incorporating contours

So far, the discussion in this paper has abstracted away from contour tones. These can also be captured with melody-local grammars with slight modifications to the `mldy` function. For example, the discussion on Prinmi in §3.2 abstracted away from word-final contours (as noted in Fn. 8). In Prinmi, it is also possible to have a word-final falling- or rising-toned syllable. Below, falling and rising tones are respectively indicated as [â] and [ǎ] on vowels and F and R in the schematic representations of strings of TBUs.

(64) Prinmi contours (Ding, 2006)

a.	bɨ̂	‘honey’	b.	tʃ’ɨ̌	‘dog’	c.	dʒjõdʒɨ̌	‘buffalo’
	F			R			LR	

Strictly speaking, in terms of strings these forms violate the generalization posited in §3.2 for Prinmi that each form must have at least one span of H-toned TBUs. However, in autosegmental terms, F and R contours are more accurately represented as HL and LH sequences, respectively, in the melody, as shown in (65) for (64c).

(65) L H

μ μ

dʒjõdʒɨ̌ ‘buffalo’ (=64c)

We can capture this with a melody-local grammar by adding a function `cntr` that 'expands' contour-toned TBUs in a string (but leaves H and L-toned TBUs as is). This encodes in our grammars the idea that, when viewed at the melody level, contours are sequences of level tones (though c.f. unit contours in some East Asian languages; Yip 1995).

(66) $\texttt{cntr}(\text{F}) = \text{HL},\ \texttt{cntr}(\text{R}) = \text{LH},\ \texttt{cntr}(\text{LR}) = \text{LLH},\ \ldots$

We can then posit that `mldy` operates on the output of `cntr`.

(67)
$$\begin{aligned}
\texttt{mldy}(\texttt{cntr}(\text{F})) &= \texttt{mldy}(\text{HL}) &&= \text{HL}\\
\texttt{mldy}(\texttt{cntr}(\text{R})) &= \texttt{mldy}(\text{LH}) &&= \text{LH}\\
\texttt{mldy}(\texttt{cntr}(\text{LR})) &= \texttt{mldy}(\text{LLH}) &&= \text{LH}
\end{aligned}$$

As shown in (67), this obtains the correct autosegmental melody for the contour-toned forms in Prinmi. Because each of the resulting strings in (67) contain exactly one H, they then conform to the melody grammar M_{Pr} for Prinmi from §3.2. Thus, adapting melody-local grammars to accomodate contours is straightforward. To what extent the `cntr` function is language specific—e.g., to what extent it should be used in East Asian tone languages—is an interesting open question.

6.3 Other representational issues

As already noted, the `mldy` function assumes the OCP in generating melody strings. Odden (1986) argues against the OCP as a hard universal; however, his arguments focus on the OCP as a constraint on underlying, lexical forms (see also Myers 1987). The only examples of OCP violation *on the surface* listed in Odden (1986) are signaled by phonetic downstep. For example, Odden lists the contrasting ARs in (68) in Kishambaa:

(68) OCP violations in Kishambaa (Odden, 1986)

a.
```
   H
  /|
nyoka 'snake'
[nyóká]
```

b.
```
 HH
 ||
ngoto 'sheep'
[ngó!tó]
```

The first, (68a) 'snake' is pronounced with two level H tones, [nyóká], and (68b) 'sheep' is pronounced with a H followed by a downstepped H; [ngó$^{!}$tó].

If surface OCP violations are indicated by downstep, then they can be detected by the `mldy` function.[14] If downstep marks are included in the inventory of tone symbols, we can modify the function such that H sequences separated by downstep are output as adjacent, yet separate H tones in the resulting melody string.

(69) $\texttt{mldy}(\text{LHH}^{!}\text{HH}) = \underset{\text{L}}{\boxed{\text{L}}}\underset{\text{H}}{\boxed{\text{H H}}}\,!\,\underset{\text{H}}{\boxed{\text{H H}}} = \text{LHH}$

Thus, OCP violations that are signaled by downstep can be straightforwardly incorporated into the current proposal. Since the ability of melody-local grammars to capture long-distance processes is based on collapsing each stretch of adjacent like-toned TBUs to

[14] Hyman (2011) lists Dioula Odienne (Braconnier, 1982) as a possible example of tautomorphemic OCP violation not marked by downstep, but he also gives an alternate, OCP-obeying analysis based on underspecification. See also Shih (2016).

a single tone in the melody, the melody-local proposal predicts that surface OCP violations should block long-distance generalizations. For example, if we imagine a situation in which OCP violations allow an unbounded number of L tones in between Hs in the melody. In these situations, the *HLH melody constraint used to capture UTP above would not be able to ban any two spans of H tones, as they would no longer always be local on the melody tier.

It should be noted that downstep does not universally indicate adjacent HH tone sequences. For instance, in Dschang-Bamileke (Tadadjeu, 1974; Hyman, 2011), a surface H$^{!}$H sequence of TBUs corresponds to a HLH sequence in the melody. The difference between Shambaa and Dschang-Bamileke shows that how the `mldy` function treats downstep must be language-specific. How a learner may discover this is thus an open question; likely, it depends on alternations (see, e.g., Pulleyblank (1986)'s extensive discussion of downstep and alternations). However, all such variations on `mldy` likely fall into the *input strictly-local* class of functions (Chandlee, 2014; Chandlee and Heinz, 2018), for which there are known learning procedures (Chandlee et al., 2014; Jardine et al., 2014). Thus, learning language-specific `mldy` functions is an open, yet approachable, question for future work.

There is also the question of underspecified TBUs—i.e., a distinction between ∅ and L-toned TBUs. This can have an effect on long-distance processes. For example, UTP is attested in Saramaccan (Roundtree 1972, p. 372; Good 2004, pp. 598–602), but there is a H/L/∅ distinction in the language, and plateauing only occurs over ∅ TBUs. In terms of the melody, *H∅H sequences are forbidden but HLH sequences are allowed. Given an inventory $\{\mathrm{H}, \mathrm{L}, \varnothing\}$, melody-local grammars can describe this pattern, and the melody-local learner can learn it. One drawback to this particular solution is that it does not quite capture the spirit of underspecification, in which unspecified TBUs would have no representation at all on the melody tier.[15] This is closer to tier projection, in which certain elements are not represented at all on the tier in question. A more direct way of incorporating underspecification may then be to combine melody and tier-projection functions.

Regardless, these methods for incorporating underspecification assume that the distinction between ∅ and L TBUs is available to the learner. Such representations are likely determined by alternations. For example, in Saramaccan the distinction between L-toned TBUs and ∅-toned TBUs is based on whether or not they show a plateauing alternation, as they are (reported to be) pronounced identically. Thus, as with downstep, learning with language-specific `mldy` functions will work in tandem with the learning of alternations (and input strictly local functions). However, here we have factored out the problem of learning representations. What has been shown here is that *if* we have the right representation, *then* we are guaranteed to learn the kind of long-distance well-formedness patterns that exist in tone. Thus, while melody-local grammars do not completely solve the learning problem in Saramaccan, they are a large step towards solving it.

There remain other aspects of representation that this paper has abstracted away from, for example the well-documented interaction between tone and metrical structure (de Lacy, 2002; Hyman, 2009). As with the other representational issues discussed in this section, this is not to deny their place in an ultimate theory of the learning of tone. Rather, the purpose of this paper has been to focus on the particular issue of learning with melodies, and thus future work can integrate this solution with other aspects of representation.

[15] Thanks to an anonymous reviewer for pointing this out.

7 Conclusion

Melody-local grammars provide a restrictive, learnable theory of tonal phonotactics based on the hypothesis that constraints on tone patterns hold either over an autosegmental melody or over local sequences of adjacent TBUs. This hypothesis was shown to be superior to conceptions of long-distance interactions based on other representational methods, including tier projection. Thus, this paper has taken an important step towards solving the learning problem by identifying a structural property of tonal patterns that can be used to learn them. This property can form the basis of further work on learning the representation of tone, learning tonal processes, and the statistical learning of tone.

References

Angluin, D. (1980). Inductive inference of formal languages from positive data. *Information and Control*, 45(2):117–135.

Beckman, J. (1998). *Positional faithfulness*. PhD thesis, University of Massachusetts, Amherst.

Bermúdez-Otero, R. (2017). Stratal phonology. In *The Routledge handbook of phonological theory*. Routledge.

Bickmore, L. S. (2007). Stem tone melodies in cilungu. *SOAS Working Papers in Linguistics*, 15:7–18.

Bickmore, L. S. and Kula, N. C. (2013). Ternary spreading and the OCP in Copperbelt Bemba. *Studies in African Linguistics*, 42.

Bickmore, L. S. and Kula, N. C. (2015). Phrasal phonology in Copperbelt Bemba. *Phonology*, 32:147–176.

Bird, S. and Ellison, T. M. (1994). One-level phonology: Autosegmental representations and rules as finite automata. *Computational Linguistics*, 20.

Braconnier, C. (1982). *Le système du dioula dOdienné, tome 1*. Institut de Linguistique Appliquée, Publication 86. Abijan, Ivory Coast: University of Abidjan, Ivory Coast.

Buckley, E. (2009). Locality in metrical typology. *Phonology*, 26:389–435.

Cassimjee, F. and Kisseberth, C. (1998). Optimal domains theory and Bantu tonology. In Kisseberth, C. and Hyman, L., editors, *Theoretical Aspects of Bantu Tone*, pages 265–314. CSLI.

Chandlee, J. (2014). *Strictly Local Phonological Processes*. PhD thesis, University of Delaware.

Chandlee, J., Eyraud, R., and Heinz, J. (2014). Learning Strictly Local subsequential functions. *Transactions of the Association for Computational Linguistics*, 2:491–503.

Chandlee, J. and Heinz, J. (2018). Strictly locality and phonological maps. *LI*, 49:23–60.

Chandlee, J., Heinz, J., and Jardine, A. (2018). Input Strictly Local opaque maps. *Phonology*, 35:1–35.

Chiośain, M. N. and Padgett, J. (2001). Markedness, segment realization, and locality in spreading. In Lombardi, L., editor, *Segmental phonology in Optimality Theory*, pages 118–156. Cambridge University Press.

Coleman, J. and Local, J. (1991). The "No Crossing Constraint" in autosegmental phonology. *Linguistics and Philosophy*, 14:295–338.

de la Higuera, C. (2010). *Grammatical Inference: Learning Automata Grammars*. Cambridge University Press.

de Lacy, P. (2002). The interaction of tone and stress in Optimality Theory. *Phonology*, 19:1–32.

Ding, P. S. (2006). A typological study of tonal systems of Japanese and Prinmi: towards a definition of pitch-accent languages. *Journal of Universal Language*, 7:1–35.

Donohue, M. (1997). Tone systems in New Guinea. *Linguistic Typology*, 1:347–386.

Eisner, J. (1997). What constraints should OT allow? Talk handout, Linguistic Society of America, Chicago. ROA#204-0797. Available at http://roa.rutgers.edu/.

Eyraud, R., Janodet, J.-C., and Oates, T. (2012). Learning substitutable binary plane graph grammars. In *Proceedings of the 11th International Conference on Grammatical Inference (ICGI 2012)*, JMLR Workshop and Conference Proceedings, pages 114–128.

Gafos, A. (1996). *The articulatory basis of locality in phonology*. PhD thesis, Johns Hopkins University.

Gallagher, G. and Wilson, C. (2018). Accidental gaps and surface-based phonotactic learning: a case study of South Bolivian Quechua. *Linguistic Inquiry*, (3):610–623.

García, P., Vidal, E., and Oncina, J. (1990). Learning locally testable languages in the strict sense. In *Proceedings of the Workshop on Algorithmic Learning Theory*, pages 325–338.

Gold, M. E. (1967). Language identification in the limit. *Information and Control*, 10:447–474.

Goldsmith, J. (1976). *Autosegmental Phonology*. PhD thesis, Massachussets Institute of Technology.

Goldsmith, J. and Riggle, J. (2012). Information theoretic approaches to phonological structure: the case of Finnish vowel harmony. *NLLT*, 30(3):859–896.

Goldsmith, J. A. (1990). *Autosegmental & Metrical Theory*. Basil Blackwell, Inc.

Good, J. (2004). Tone and accent in Saramaccan: charting a deep split in the phonology of a language. *Lingua*, 114:575–619.

Gouskova, M. and Gallagher, G. (to appear). Inducing nonlocal constraints from baseline phonotactics. *Natural Language and Linguistic Theory*.

Graf, T. (2017). The power of locality domains in phonology. *Phonology*, 34:385–405.

Haraguchi, S. (1977). *The Tone Pattern of Japanese: An Autosegmental Theory of Tonology*. Tokyo: Kaitakusha.

Hayes, B. and Wilson, C. (2008). A maximum entropy model of phonotactics and phonotactic learning. *LI*, 39:379–440.

Heinz, J. (2010a). Learning long-distance phonotactics. *LI*, 41:623–661.

Heinz, J. (2010b). String extension learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 897–906. Association for Computational Linguistics.

Heinz, J., de la Higuera, C., and van Zaanen, M. (2016). *Grammatical Inference for Computational Linguistics*. Number 28 in Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Heinz, J., Rawal, C., and Tanner, H. G. (2011). Tier-based strictly local constraints for phonology. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 58–64, Portland, Oregon, USA. Association for Computational Linguistics.

Heinz, J. and Rogers, J. (2010). Estimating strictly piecewise distributions. In *Proceedings of the 48th Annual Meeting of the ACL*. Association for Computational Linguistics.

Heinz, J. and Rogers, J. (2013). Learning subregular classes of languages with factored deterministic automata. In Kornai, A. and Kuhlmann, M., editors, *Proceedings of the 13th Meeting on the Mathematics of Language (MoL 13)*, pages 64–71, Sofia, Bulgaria. Association for Computational Linguistics.

Hewitt, M. and Prince, A. (1989). OCP, locality, and linking: the N. Karanga verb. In Fee, E. J. and Hunt, K., editors, *WCCFL 8*, pages 176–191. Stanford: CSLI Publications.

Hirayama, T. (1951). *Kyuusyuu hoogen Onchoo no Kenkyuu (Studies on the Tone of the Kyushu Dialects)*. Tokyo: Gakkai no shinshin-sha.

Hopcroft, J., Motwani, R., and Ullman, J. (2006). *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley, third edition.

Hyde, B. (2012). Alignment constraints. *NLLT*, 30:789–836.

Hyman, L. (2001). Tone systems. In Haspelmath, M., nig, E. K., Oesterreicher, W., and Raible, W., editors, *Language typology and language universals: an international handbook*, volume 2, pages 1367–1380. Berlin & New York: Walter de Gruyter.

Hyman, L. (2007). Universals of tone rules: 30 years later. In *Tones and Tunes Volume 1: Typological studies in Word and Sentence Prosody*, pages 1–34. New York: Mouton de Gruyter.

Hyman, L. (2009). How not to do typology: the case of pitch-accent. *Language Sciences*, pages 213–238.

Hyman, L. (2011). Tone: Is it different? In Goldsmith, J. A., Riggle, J., and Yu, A. C. L., editors, *The Blackwell Handbook of Phonological Theory*, pages 197–238. Wiley-Blackwell.

Hyman, L. and Katamba, F. X. (1993). A new approach to tone in luganda. *Language*, 69:34–67.

Hyman, L. and Katamba, F. X. (2010). Tone, syntax and prosodic domains in Luganda. In Downing, L., Rialland, A., Beltzung, J.-M., Manus, S., Patin, C., and Riedel, K., editors, *Papers from the Workshop on Bantu Relative Clauses*, volume 53 of *ZAS Papers in Linguistics*, pages 69–98. ZAS Berlin.

Jardine, A. (2016). Computationally, tone is different. *Phonology*, 33:247–283.

Jardine, A. (2017). The local nature of tone-association patterns. *Phonology*, 34:363–384.

Jardine, A. (2019). The expressivity of autosegmental grammars. *Journal of Logic, Language, and Information*, 28:9–54.

Jardine, A., Chandlee, J., Eyraud, R., and Heinz, J. (2014). Very efficient learning of structured classes of subsequential functions from positive data. In *Proceedings of the 12th International Conference on Grammatical Inference (ICGI 2014)*, JMLR Workshop Proceedings, pages 94–108.

Jardine, A. and Heinz, J. (2015). A concatenation operation to derive autosegmental graphs. In *Proceedings of the 14th Meeting on the Mathematics of Language (MoL 2015)*, pages 139–151, Chicago, USA. Association for Computational Linguistics.

Jardine, A. and Heinz, J. (2016). Learning tier-based strictly 2-local languages. *Transactions of the Association for Computational Linguistics*, 4:87–98.

Jardine, A. and McMullin, K. (2017). Efficient learning of tier-based strictly k-local languages. In Drewes, F., Martín-Vide, C., and Truthe, B., editors, *Language and Automata Theory and Applications, 11th International Conference*, Lecture Notes in Computer Science, pages 64–76. Springer.

Jurafsky, D. and Martin, J. H. (2009). *Speech and Language Processing*. Pearson Prentice Hall, 2nd edition edition.

Kiparsky, P. (2000). Opacity and cyclicity. *Linguistic Review*, 17:351–366.

Kisseberth, C. and Odden, D. (2003). Tone. In Nurse, D. and Philippson, G., editors, *The Bantu Languages*. New York: Routledge.

Kornai, A. (1995). *Formal Phonology*. Garland Publication.

Kubozono, H. (2012). Varieties of pitch accent systems in Japanese. *Lingua*, 122:1395–1414.

Lai, R. (2015). Learnable versus unlearnable harmony patterns. *LI*, 46:425–451.

Leben, W. R. (1973). *Suprasegmental phonology*. PhD thesis, Massachussets Institute of Technology.

McCarthy, J. J. (1985). *Formal Problems in Semitic Phonology and Morphology*. New York: Garland.

McMullin, K. and Hansson, G. O. (2015). Locality in long-distance phonotactics: evidence for modular learning. In *NELS 44, Amherst, MA*. in press.

McNaughton, R. and Papert, S. (1971). *Counter-Free Automata*. MIT Press.

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, 63(2):81–97.

Myers, S. (1987). *Tone and the structure of words in Shona*. PhD thesis, University of Massachusetts, Amherst.

Myers, S. (1997). OCP effects in Optimality Theory. *NLLT*, 15(4):847–892.

Odden, D. (1982). Tonal phenomena in Kishambaa. *Studies in African Linguistics*, 13(2):177–208.

Odden, D. (1984). Stem tone assignment in Shona. In Clements, G. and Goldsmith, J. A., editors, *Autosegmental Studies in Bantu Tone*, pages 255–280. Foris Publications.

Odden, D. (1986). On the role of the Obligatory Contour Principle in phonological theory. *Language*, 62(2):353–383.

Odden, D. (1994). Adjacency parameters in phonology. *Language*, 70(2):289–330.

Pater, J. and Tessier, A.-M. (2003). Phonotactic knowledge and the acquisition of alternations. In Solé, M.-J., Recasens, D., and Romero, J., editors, *Proceedings of the 15th International Congress of Phonetic Sciences*, pages 1177–1180. Barcelona: Universitat Autónoma de Barcelona.

Pierrehumbert, J. B. and Beckman, M. E. (1988). *Japanese Tone Structure*. MIT Press, Cambridge, MA.

Pulleyblank, D. (1986). *Tone in Lexical Phonology*. Dordrecht: D. Reidel.

Rogers, J., Heinz, J., Fero, M., Hurst, J., Lambert, D., and Wibel, S. (2013). Cognitive and sub-regular complexity. In *Formal Grammar*, volume 8036 of *Lecture Notes in Computer Science*, pages 90–108. Springer.

Rogers, J. and Pullum, G. (2011). Aural pattern recognition experiments and the subregular hierarchy. *Journal of Logic, Language and Information*, 20:329–342.

Roundtree, S. C. (1972). Saramaccan tone in relation to intonation and grammar. *Lingua*, 29:308–325.

Shih, S. and Inkelas, S. (forthcoming). Autosegmental aims in surface optimizing phonology. Ms. available at lingbuzz/002520.

Shih, S. S. (2016). Super additive similarity in dioula tone harmony. In Kim, K., Umbal, P., Block, T., Chan, Q., Cheng, T., Finney, K., Katz, M., Nickel-Thompson, S., and Shorten, L., editors, *Proceedings of the 33rd West Coast Conference on Formal Linguistics*, pages 361–370.

Tadadjeu, M. (1974). Floating tones, shifting rules, and downstep in Dschang-Bamileke. In Leben, W. R., editor, *Papers from the 5th Annual Conference on African Linguistics*, Studies in African Linguistics, Supplement 5.

Williams, E. S. (1976). Underlying tone in Margi and Igbo. *Linguistic Inquiry*, 7(3):463–484.

Yip, M. (1988). Template morphology and the direction of association. *NLLT*, 6:551–577.

Yip, M. (1995). Tone in East Asian languages. In *The handbook of phonological theory*, pages 476–494. Oxford: Blackwell.

Yip, M. (2002). *Tone*. Cambridge University Press.

Zoll, C. (2003). Optimal tone mapping. *LI*, 34(2):225–268.

Appendix

This appendix collects the formal details of the paper. Standard notation for set theory is used. Let Σ be a fixed, finite alphabet of symbols and Σ^* be the set of all strings over Σ, including λ, the empty string. For a symbol $\sigma \in \Sigma$, σ^n denotes the string resulting from n repetitions of σ. Let $|w|$ indicate the length of a string w. For two strings $w, v \in \Sigma^*$, let wv denote their concatenation (likewise for $w \in \Sigma^*$ and $\sigma \in \Sigma$, $w\sigma$ denotes their concatenation). A *stringset* (or formal language) is a subset of Σ^*; this corresponds to the notion of pattern discussed in §2.1. Let $\rtimes$ and $\ltimes$ represent special boundary symbols not in Σ that represent the beginning and end of words, respectively; thus, $\rtimes w \ltimes$ is the string w delineated with the boundary strings. (These correspond to the # boundary used in phonology.)

A.1 Strictly local grammars and k-factors

A string u is a *k-factor* of w if $|u| = k$ and $w = v_1 u v_2$ for some $v_1, v_2 \in \Sigma^*$; that is, u is a substring of w of length k. The *k-factors* of w are given by the following function $\texttt{fac}_k$:

$$\texttt{fac}_k(w) \stackrel{\text{def}}{=} \begin{cases} \{u \mid u \text{ is a } k\text{-factor of } \rtimes w \ltimes\} & \text{if } |\rtimes w \ltimes| > k \\ \{\rtimes w \ltimes\} & \text{otherwise} \end{cases}$$

For instance, $\texttt{fac}_3(\text{LHLL}) = \{\rtimes\text{LH}, \text{LHL}, \text{HLL}, \text{LL}\ltimes\}$. We extend $\texttt{fac}_k$ to stringsets in the natural way; i.e. for $L \subseteq \Sigma^*$, $\texttt{fac}_k(L) = \bigcup_{w \in L} \texttt{fac}_k(w)$.

A strictly k-local (SL$_k$) grammar is a set $S \subseteq \texttt{fac}_k(\Sigma^*)$; that is, a subset of all of the possible k-factors that can appear in strings in S. For example, for $\Sigma = \{\text{L}, \text{H}\}$,

$$\texttt{fac}_2(\Sigma^*) = \{\rtimes\text{H}, \rtimes\text{L}, \text{HH}, \text{HL}, \text{LH}, \text{LL}, \text{H}\ltimes, \text{L}\ltimes\}.$$

Then, for example, $S_{\text{alt}} = \{\rtimes\text{H}, \text{HH}, \text{LL}, \text{L}\ltimes\}$ is a SL$_2$ grammar because $\rtimes$H, HH, LL, and L$\ltimes$ are all 2-factors of strings in Σ^* (for example, they are all in $\texttt{fac}_2(\text{HHLL})$).

The for a SL$_k$ grammar S, the stringset described by S, written $L(S)$, is thus the set of strings that contain no k-factors in S; that is,

$$L(S) \stackrel{\text{def}}{=} \{w \in \Sigma^* \mid \texttt{fac}_k(w) \cap S = \emptyset\}$$

For example,

$$L(S_{\text{alt}}) = \{\text{LH}, \text{LHLH}, \text{LHLHLH}, ...\},$$

that is, the set of strings of alternating Hs and Ls, as this is exactly the set of strings that contain none of the 2-factors in S_{alt}.

A stringset L is thus strictly k-local iff $L = L(S)$ for some SL$_k$ grammar S. We say a stringset is strictly local if it is strictly k-local for some k.

The learning procedure for the class of strictly k-local stringsets amounts to the function $\texttt{SLlearn}_k$ defined as follows. For a finite set $D \subset \Sigma^*$,

$$\texttt{SLlearn}_k(D) \stackrel{\text{def}}{=} \texttt{fac}_k(\Sigma^*) - \texttt{fac}_k(D)$$

That is, $\texttt{SLlearn}_k$(D) returns the set of possible k-factors minus the set of k-factors observed in D. This means that $\texttt{SLlearn}_k$ returns a strictly k-local grammar consisting of all of the k-factors *not* observed in D. It should be noted that this is a 'batch' conception of the learner, as opposed to the sequential learner presented in the main text. They are equivalent, however.

The sequential version of the learner takes some finite sequence of data points d_1, d_2, d_3, ..., d_n and returns, at each data point d_i, $\texttt{SLlearn}_k(\{d_1, d_2, d_3, ..., d_i\})$.

The following theorem asserts the correctness of $\texttt{SLlearn}_k$.

Theorem 1 *For a target strictly k-local stringset L and a sample D of L such that* $\texttt{fac}_k(D) = \texttt{fac}_k(L)$, $\texttt{SLlearn}_k(D)$ *returns a strictly k-local grammar S such that* $L(S) = L$.

Proof We show first that $w \in L$ implies $w \in L(S)$ and then that $w \in L(S)$ implies $w \in L$. Since $\texttt{fac}_k(D) = \texttt{fac}_k(L)$, then $\texttt{fac}_k(\Sigma^*) - \texttt{fac}_k(D) = \texttt{fac}_k(\Sigma^*) - \texttt{fac}_k(L)$. Since $S = \texttt{SLlearn}_k(D) = \texttt{fac}_k(\Sigma^*) - \texttt{fac}_k(D)$, then $S = \texttt{fac}_k(\Sigma^*) - \texttt{fac}_k(L)$. Thus for every $w \in L$, $\texttt{fac}_k(w) \cap S = \emptyset$, so $w \in L(S)$.

Because L is a strictly k-local set, there is some strictly k-local grammar S' such that $L(S') = L$. Note that for any string w that if $\texttt{fac}_k(w) \subseteq \texttt{fac}_k(L)$, then $\texttt{fac}_k(w) \cap S' = \emptyset$, and so $\texttt{fac}_k(w) \in L$. Because $S = \texttt{fac}_k(\Sigma^*) - \texttt{fac}_k(L)$. For $w \in L(S)$, then $\texttt{fac}_k(w) \cap (\texttt{fac}_k(\Sigma^*) - \texttt{fac}_k(L)) = \emptyset$ and so $\texttt{fac}_k(w) \subseteq \texttt{fac}_k(L)$. Thus $w \in L(S)$ implies that $w \in L$.

A.2 Melody-local grammars and their learning

Having defined strictly local stringsets and their learning, we can now define melody-local stringsets.

First, we define the $\texttt{mldy}$ function recursively as follows. For $w \in \Sigma^*$,

$$\texttt{mldy}(w) \stackrel{\text{def}}{=} \begin{cases} \lambda & \text{if } w = \lambda, \\ \texttt{mldy}(v)\sigma & \text{if } w = v\sigma^n,\ v \neq u\sigma \text{ for some } u \in \Sigma^* \end{cases}$$

That is, $\texttt{mldy}(w)$ returns λ if $w = \lambda$, otherwise it returns $\texttt{mldy}(v)\sigma$, where v is the longest string not ending in σ. For example,

$$\begin{aligned} \texttt{mldy}(\text{HHLLLH}) &= \texttt{mldy}(\text{HHLLL})\text{H} \\ &= \texttt{mldy}(\text{HH})\text{LH} \\ &= \texttt{mldy}(\lambda)\text{HLH} \\ &= \lambda\text{HLH} = \text{HLH} \end{aligned}$$

For a stringset $L \subseteq \Sigma^*$ let $\texttt{mldy}(L) = \{\texttt{mldy}(w) \mid w \in L\}$.

A melody strictly k-local grammar M is thus, like a strictly k-local grammar, a subset of the possible k factors of Σ. That is, $M \subseteq \texttt{fac}_k(\Sigma^*)$. The difference is that we interpret a melody strictly k-local grammar using the $\texttt{mldy}$ function. The stringset described by M is as follows:

$$L(M) \stackrel{\text{def}}{=} \{w \in \Sigma^* \mid \texttt{fac}_k(\texttt{mldy}(w)) \cap M = \emptyset\}$$

Thus, for example, if $k = 3$ and $M = \{\text{HLH}\}$, then HHLLLH $\notin L(M)$, because $\texttt{mldy}(\text{HHLLLH}) = \text{HLH}$ and $\texttt{fac}_k(\text{HLH}) \cap M = \{\text{HLH}\}$. However, HLLLL$\in L(M)$, because $\texttt{mldy}(\text{HLLLL}) = \text{HL}$ and $\texttt{fac}_k(\text{HL}) \cap M = \emptyset$.

We can then define a k, j-melody-local grammar G as a tuple $G(S, M)$ where S is a strictly k-local grammar and M is a melody strictly j-local grammar. The stringset described by G is thus

$$L(G) \stackrel{\text{def}}{=} L(S) \cap L(M),$$

that is, the set of strings that satisfy both S and M. We say a stringset is melody-local if it is k, j-melody-local for some k and j.

Learning melody-local stringsets is a straightforward extension of learning strictly local stringsets. If we fix k, we can define a learning function that takes an input D and outputs the following result:

$$\mathtt{MLlearn}_{k,j}(D) \stackrel{\text{def}}{=} \left(\mathtt{SLlearn}_k(D), \mathtt{SLlearn}_j(\mathtt{mldy}(D))\right)$$

That is, $\mathtt{MLlearn}_{k,j}(D)$ returns a tuple, the first of which is obtained by running a strictly k-local learning on D, the second of which is a melody strictly j-local grammar obtained by running strictly j-local learning on $\mathtt{mldy}(D)$. The following theorem asserts the correctness of $\mathtt{MLlearn}_{k,j}$.

Theorem 2 *For a target k, j-melody-local stringset L and a sample D of L such that $\mathtt{fac}_k(D) = \mathtt{fac}_k(L)$ and $\mathtt{fac}_j(\mathtt{mldy}(D)) = \mathtt{fac}_j(\mathtt{mldy}(L))$, $\mathtt{MLlearn}_{k,j}(D)$ returns a k, j-melody-local grammar G such that $L(G) = L$.*

Proof Almost immediate from Thm. 1. If L is k-melody-local, then there is some k-melody-local grammar $G' = (S', M')$ such that $L(G') = L$. Let $G = (S, M)$. Because $\mathtt{fac}_k(D) = \mathtt{fac}_k(L)$ and $\mathtt{fac}_j(\mathtt{mldy}(D)) = \mathtt{fac}_j(\mathtt{mldy}(L))$, from Thm. 1 we know that $L(S) = L(S')$ and $L(M) = L(M')$. Thus $L(G) = L(G') = L$.

A.3 Abstract Characterization

We can posit an abstract characterization for melody-local patterns independent of a particular grammar formalism to describe them. This allows us to prove whether or not a pattern is melody-local. We base this off of the abstract characterization of strictly local stringsets. Strictly local stringsets can be characterized by the property of *suffix substitution closure* (Rogers and Pullum, 2011; Rogers et al., 2013), which can be used to prove that a pattern is not strictly local.

Theorem 3 (Suffix substitution closure (Rogers and Pullum, 2011)) *A stringset L is SL_k iff for any string x of length $k-1$ and any strings u_1, u_2, w_1, and w_2,*

$$\text{if } u_1xu_2 \in L \text{ and } w_1xw_2 \in L, \text{ then } u_1xw_2 \in L$$

This means that, for any $u_1xu_2 \in L$, and for any $w_1xw_2 \in L$, then, as long as x is of length $k-1$, then we can freely replace u_2 with w_2 and be guaranteed to produce another string in L. For example, for the stringset L_{KJ} (penultimate or final H tone) from the main text, we can set x to be HL (because $k = 3$, x must be of length 2), and u_1, u_2, w_1, and w_2 as in (71).

(70) $L_{\text{KJ}} = \{$H, HL, LHL, LLHL, LLLHL, ..., LH, LLH, LLLH, ...$\}$

(71)
$$\begin{array}{l}
\underbrace{\text{LLLL}}_{u_1}\,\underbrace{\text{LH}}_{x}\,\underbrace{\lambda}_{u_2} \in L_{\text{KJ}} \\
\underbrace{\text{L}}_{w_1}\,\underbrace{\text{LH}}_{x}\,\underbrace{\text{L}}_{w_2} \in L_{\text{KJ}} \\
\hline
\underbrace{\text{LLLL}}_{u_1}\,\underbrace{\text{LH}}_{x}\,\underbrace{\text{L}}_{w_2} \in L_{\text{KJ}}\ \checkmark
\end{array}$$

Thus, u_1xu_2 is LLLLLH, which is a member of L_{KJ}, and w_1xw_2 is LLHL, which is also a member of L_{KJ}. If we substitute u_2 for w_2 in the former, then we obtain a new string $u_1xw_2 =$ LLLLLHL, which is also in L_{KJ}. We can do this for any x of length 2. Another example is given below in (72) for $x =$ LL.

(72) $$\begin{array}{l} \underbrace{\text{LLLL}}_{u_1}\ \underbrace{\text{LL}}_{x}\ \underbrace{\text{HL}}_{u_2} \in L_{\text{KJ}} \\ \underbrace{\text{L}}_{w_1}\ \underbrace{\text{LL}}_{x}\ \underbrace{\text{LLLHL}}_{w_2} \in L_{\text{KJ}} \\ \hline \underbrace{\text{LLLL}}_{u_1}\ \underbrace{\text{LL}}_{x}\ \underbrace{\text{LLLHL}}_{w_2} \in L_{\text{KJ}} \ \checkmark \end{array}$$

To show that a stringset is *not* strictly local, we show that suffix substitution closure fails for some x no matter the size of k. Recall the stringset L_{Ch} (at least one H) from the main text.

(73) $L_{\text{Ch}} = \{\text{H, HL, LH, HH, LLH, LHL, LHH, HLL, HLH, HHH, LLLH, ...}\}$

If, as in L_{KJ}, we set $k = 3$ and choose the string LL, then L_{Ch} fails suffix substitution closure for $x = \text{LL}$ and u_1, u_2, w_1, w_2 chosen as shown in (74).

(74) $$\begin{array}{l} \underbrace{\text{L}}_{u_1}\ \underbrace{\text{LL}}_{x}\ \underbrace{\text{H}}_{u_2} \in L_{\text{Ch}} \\ \underbrace{\text{H}}_{w_1}\ \underbrace{\text{LL}}_{x}\ \underbrace{\text{L}}_{w_2} \in L_{\text{Ch}} \\ \hline \underbrace{\text{L}}_{u_1}\ \underbrace{\text{LL}}_{x}\ \underbrace{\text{L}}_{w_2} \notin L_{\text{Ch}} \ \text{✗} \end{array}$$

Because $u_1xw_2 = \text{LLLL}$ is not a member of L_{Ch}, L_{Ch} is not strictly 3-local. Furthermore, there is no k for which L_{Ch} is strictly k-local, because we can simply replace x with L^{k-1} ($k-1$ repetitions of L).

(75) $$\begin{array}{l} \underbrace{\text{L}}_{u_1}\ \underbrace{\text{L}^{k-1}}_{x}\ \underbrace{\text{H}}_{u_2} \in L_{\text{Ch}} \\ \underbrace{\text{H}}_{w_1}\ \underbrace{\text{L}^{k-1}}_{x}\ \underbrace{\text{L}}_{w_2} \in L_{\text{Ch}} \\ \hline \underbrace{\text{L}}_{u_1}\ \underbrace{\text{L}^{k-1}}_{x}\ \underbrace{\text{L}}_{w_2} \notin L_{\text{Ch}} \ \text{✗} \end{array}$$

This shows that, no matter what $k-1$, suffix substitution in this case will produce a string LL^{k-1}L, which is not a member of L_{Ch}. Thus, L_{Ch} fails suffix substitution closure for any k. This is a formal version of the intuitive ‘scanning’ proof given in §2.3, (13).

From the suffix substitution closure characterization of strictly local stringsets, we can posit *melody-dependent suffix substitution closure* as the abstract characterization of melody-local stringsets.

Theorem 4 (Melody-dependent suffix substitution closure (MSSC)) *A stringset L is melody-local iff, for some k and some j,*

1. $\texttt{mldy}(L)$ *is strictly j-local and*
2. *for any strings* w_1, w_2, u_1, u_2 *and for any string x,* $|x| = k-1$,

$$w_1xw_2 \in L \text{ and } u_1xu_2 \in L \text{ and } \texttt{mldy}(w_1xu_2) \in \texttt{mldy}(L) \text{ implies } w_1xu_2 \in L$$

Proof Recall that a stringset is melody-local iff it is describable by some melody-local grammar $G = (S, M)$. Thm. 4a follows directly from the definition of $L(M)$. Thm. 4b follows from suffix substitution closure for $L(S)$ plus the additional requirement that $L(G) = L(S) \cap L(M)$.

Melody-dependent suffix substitution closure adds two conditions on suffix substitution closure. First, Thm. 4a states that $\texttt{mldy}(L)$ (the stringset consisting of the melodies of all strings in L) must be strictly j-local. Second, Thm. 4b adds to the antecedent of the suffix substitution closure implication that $\texttt{mldy}(w_1xu_2)$ must be in $\texttt{mldy}(L)$. As an example, take L_{Ch}. First, note that $\texttt{mldy}(L_{\text{Ch}})$ (given below in (76)), is strictly 3-local, as witnessed by the melody strictly j-local grammar $M_{\text{Ch}} = \{\rtimes \text{L} \ltimes\}$ (i.e., it does not contain the string L).

(76) $\texttt{mldy}(L_{\text{Ch}}) = \{\text{H}, \text{HL}, \text{LH}, \text{HLH}, \text{LHL}, \text{HLHL}, \ldots\}$

It is also then true that L_{Ch} satisfies Thm. 4 for $k = j = 3$. While L_{Ch} fails the implication in (74) for suffix substitution closure, this implication holds for melody-dependent suffix substitution closure, because $\texttt{mldy}(\text{LLLL}) = \text{L}$ is not a member of $\texttt{mldy}(L_{\text{Ch}})$, and so it does not matter that LLLL $\notin L_{\text{Ch}}$.

(77)
$$\begin{array}{l}
\underbrace{\text{L}}_{u_1}\underbrace{\text{LL}}_{x}\underbrace{\text{H}}_{u_2} \in L_{\text{Ch}} \\
\underbrace{\text{H}}_{w_1}\underbrace{\text{LL}}_{x}\underbrace{\text{L}}_{w_2} \in L_{\text{Ch}} \\
\texttt{mldy}(\underbrace{\text{L}}_{u_1}\underbrace{\text{LL}}_{x}\underbrace{\text{L}}_{w_2}) \notin \texttt{mldy}(L_{\text{Ch}}) \checkmark \\
\hline
\underbrace{\text{L}}_{u_1}\underbrace{\text{LL}}_{x}\underbrace{\text{L}}_{w_2} \notin L_{\text{Ch}}
\end{array}$$

It is thus the case that L_{Ch} satisfies melody-dependent suffix substitution closure.

To give an example that does not, recall the 'no consecutive spreading Hs' pattern discussed in §5. More explicitly, this is the set L_{No2H} as follows.

(78) L_{No2H} is exactly the set that is the union of

1. The set of all L strings; {L, LL, LLL, LLLL ...}
2. The set of strings containing a single H span (i.e. L_{UTP}):
   $L_{\text{UTP}} = \{$ L, H, LL, LH, HL, HH, LLL, LLH, LHL, LHH, HLL, HHL, HHH, LLLL, LLLH, LLHL, LLHH, LHLL, LHHL, LHHH, HLLL, HHLL, HHHL, ... }
3. The set of strings containing more than one H span, where given any two consecutive H spans, *only one can be more than one TBU long*:
   {HLH, HLLH, HHLH, HLHH, HLLLH, HLHHH, HHHLH, HLLLLH, ... }

That is, L_{No2H} is exactly the set *not* containing any strings like *HHLLHH, or *HHHLHH, or *HHLLLHH, where H spans separated by exactly one L-span are *both* of more than one TBU.

There are no constraints on the melody in this pattern; thus $\texttt{mldy}(L_{\text{No2H}})$ is the full set of alternating strings of Hs and Ls.

(79) $\texttt{mldy}(L_{\text{No2H}}) = \{\text{H}, \text{L}, \text{HL}, \text{LH}, \text{HLH}, \text{LHL}, \text{HLHL}, \ldots\}$

We can show that this fails melody-dependent substitution closure using example strings based on the ARs in (63) from the main text.

(80)
$$
\begin{array}{r}
\underbrace{\mathrm{HHHH}}_{u_1}\,\underbrace{\mathrm{L}^{k-1}}_{x}\,\underbrace{\mathrm{H}}_{u_2} \in L_{\mathrm{No2H}} \\
\underbrace{\mathrm{H}}_{w_1}\,\underbrace{\mathrm{L}^{k-1}}_{x}\,\underbrace{\mathrm{HHHH}}_{w_2} \in L_{\mathrm{No2H}} \\
\texttt{mldy}(\underbrace{\mathrm{HHHH}}_{u_1}\,\underbrace{\mathrm{L}^{k-1}}_{x}\,\underbrace{\mathrm{HHHH}}_{w_2}) \in \texttt{mldy}(L_{\mathrm{No2H}}) \\
\hline
\underbrace{\mathrm{HHHH}}_{u_1}\,\underbrace{\mathrm{L}^{k-1}}_{x}\,\underbrace{\mathrm{HHHH}}_{w_2} \notin L_{\mathrm{No2H}} \quad ✗
\end{array}
$$

In this case, $u_1xw_2 = \mathrm{HHHHL}^{k-1}\mathrm{HHHH}$, in which two consecutive H spans have spread more than two TBUs (as in (632) in the main text). This satisfies the melody constraint (because, e.g., HHHHLH$\in L_{\mathrm{No2H}}$ and so HLH$\in \texttt{mldy}(L_{\mathrm{No2H}})$), but it is not in L_{No2H}, so it fails the implication, for any k. Thus, 'the no consecutive spreading Hs' pattern L_{No2H} is not melody-local.