The Power of Locality Domains in Phonology

Abstract

Locality domains play an integral role in linguistic theories. This paper combines locality domains with current work on the computational complexity of phonology. The general upshot is that if a specific formalism — the strictly piecewise grammars — is supplemented with a mechanism to enforce domain restrictions, its power increases so much that it subsumes almost the full hierarchy of subregular languages. However, if domain restrictions must follow a linguistically natural template, one instead obtains a much more restricted class that unifies all phonologically natural subregular classes. At the same time, unnatural abilities of some subregular classes such as counting and elaborate conditionals are correctly excluded. In addition, this restricted class is learnable in the limit from positive data.

Keywords: locality, subregular hierarchy, strictly piecewise, learnability, word domain

word count according to texcount: 7011

1 Introduction

It has been known for a long time that the phonological systems of natural languages are finite-state in nature and thus generate regular languages (Johnson 1972; Kaplan and Kay 1994). However, this characterization is too loose in the sense that phonological dependencies instantiate much simpler patterns — the regular languages only provide an upper bound on phonological expressivity, and a very generous one at that. This realization has led to increased efforts in recent years to pinpoint the location of phonology more precisely (Heinz 2009, 2010; Graf 2010a,b; Rogers et al. 2013; Jardine 2015). The subregular hierarchy, first developed in McNaughton and Pappert (1971), provides a fine-grained selection of language classes that are less complex than regular languages. Phonological considerations have prompted the addition of various classes to this hierarchy (Rogers et al. 2010; Heinz et al. 2011), and a tentative consensus has emerged that the classes of strictly local, strictly piecewise, and tier-based strictly local languages — and the grammars that generate them — provide reasonable approximations of natural language phonotactics.

So far, though, these investigations have implicitly assumed that these grammars only need to regulate the well-formedness of individual words and that multi-word strings are not proper phonological objects. This is problematic for two reasons. First, it completely precludes the study of processes or constraints that apply across word boundaries. Second, Graf (2010b) already points out that the complexity of some processes such as Sanskrit n-retroflexion depends on whether the phonological objects generated by the grammar are single words or sequences of words, sentences or at least parts of sentences.

This paper takes the issue of words versus sentences as its vantage point for a more general investigation of the impact of locality domains in phonology. I first corroborate Graf's observation by showing that some processes that over words belong to one of the phonologically reasonable subregular classes cannot be captured if the domain is widened to include sequences of words. An interesting solution to this problem is offered by the novel class of *domain-based strictly piecewise grammars*, a minor extension of strictly piecewise grammars that allows them to define locality domains within which specific phonotactic principles must be obeyed. It turns out that all subregular classes can be instantiated by picking specific locality domains, with the three phonologically natural subregular classes corresponding to the special case where the locality domains obey a simple and highly intuitive template. By restricting the class of locality domains to phonologically natural ones, one thus obtains a formalism that can handle all attested phonotactic patterns while excluding the empirically uninteresting parts of the subregular hierarchy. In addition, the restriction to natural locality domains guarantees learnability in the limit from positive text. All of this lends credence to the long-held belief that locality domains serve an integral role in shaping phonotactics and aid language acquisition.

The paper is laid out as follows. Section 2 familiarizes the reader with a variety of subregular classes. The empirically viable classes of strictly local, strictly piecewise and tier-based strictly local grammars are covered in Sec. 2.1–2.3. Section 2.3 also discusses the strictly threshold testable, strictly co-threshold testable, and locally testable grammars. While these can capture some aspects of phonotactics, they mostly serve as examples of what kind of mechanisms should not be available in any reasonable model of phonology. This point is further emphasized in 2.4, which shows that these unnatural grammars break down once the domain is widened from words to sentences. Among the natural grammar formalisms, on the other hand, only the strictly piecewise grammars struggle in this case.

In order to fix this issue, section 3.1 then extends strictly piecewise grammars to domain-based strictly piecewise grammars. A particularly interesting subtype are the interval-based strictly piecewise grammars, which subsume the strictly local and tier-based strictly local grammars (3.2 and 3.3), but not the strictly (co-)threshold testable or locally testable grammars (3.4). Section 3.5 concludes with a proof that even though the domain-based strictly piecewise languages are not learnable, the interval-based ones are thanks to their more limited notion of locality domain.

A minor caveat is in order before we proceed to the main body of the paper. While a lot of recent work in the literature has focused on the computational complexity of the mapping from underlying forms to surface forms (see Chandlee 2014 and references therein), this paper restricts itself to phonotactics, i.e. surface forms. The formalisms discussed here only need to be able to distinguish well-formed surface forms from ill-formed ones, underlying representations are never considered. Nonetheless I will sometimes describe certain phenomena such as tone plateauing in terms of rewrite processes. This is merely a convenient expository device, the only relevant issue for this paper is the surface forms these processes produce.

2 Subregular Phonotactics

2.1 Strictly Local Grammars

The simplest model of phonotactics is provided by strictly local grammars. A strictly k-local grammar is a finite set of k-grams, each one of which is a description of an illicit sequence. For example, the strictly 2-local grammar $\{\rtimes\mathrm{V}, \mathrm{CC}, \mathrm{VV}, \mathrm{C}\ltimes\}$ forbids any word to start with a vowel ($\rtimes$ indicates the left edge of the string), end in a consonant ($\ltimes$ indicates the right edge of the string), or contain a sequence of two vowels or two consonants. In other words, this grammar describes a language with a CV syllable template.

Note that C and V are used for the sake of brevity here, a realistic grammar would have to replace these symbols by the corresponding phones of the language. So if the language in question has only the vowels [a], and [u], plus two consonants [p] and [k], the set of sequences that violate the CV template would have to be compiled out accordingly.

⋊a	pp	aa	p⋉
⋊u	pk	au	k⋉
	kp	ua	
	kk	uu	

We can increase the length of the k-grams to capture processes with an extended locality domain. Consider the case of intervocalic voicing of the fricatives /f/ and /s/. Since the model only considers phonotactics, the task is not to regulate the realization of underlying /f/ and /s/ but to block the occurrence of the phones [f] and [s] between vowels. This is exactly what the strictly 3-local grammar {VfV, VsV} accomplishes (once V has been replaced by the actual vowels of the language). A strictly 2-local grammar is insufficient, as it would have to incorrectly deem Vf and fV to be illicit sequences. Since either one is fine as long as [f] is not followed/preceded by another vowel, the grammar would be too restrictive.

Strictly local grammars can be recast in terms of formulas of first-order logic, which will prove to be a very useful perspective for the remainder of the paper, in particular Sec. 3. I refrain from giving a full exposition of first-order logic here — a purely intuitive treatment suffices for our purposes. One may think of the grammar for the CV-template given above as a single constraint of the following form: for any two segments x_1 and x_2 of the string it holds that if x_1 is immediately followed by x_2, then I) it is not the case that x_1 is labeled ⋊ and x_2 is labeled V, and II) it is not the case that x_1 is labeled C and x_2 is labeled C, and III) it is not the case that x_1 is labeled V and x_2 is labeled V, and IV) it is not the case that x_1 is labeled C and x_2 is labeled ⋉. This statement can be directly translated into first-order logic using $x_1\ S\ x_2$ to indicate that x_2 immediately follows x_1 (in mathematical parlance, "x_1 is the successor of x_2").

$$\forall x_1, x_2 \Big[x_1\ S\ x_2 \rightarrow \neg(\rtimes(x_1) \wedge V(x_2)) \wedge \neg(C(x_1) \wedge C(x_2)) \wedge \neg(V(x_1) \wedge V(x_2)) \wedge \neg(C(x_1) \wedge \ltimes(x_2)) \Big]$$

2.2 Strictly Piecewise Grammars

Strictly local grammars — despite their simplicity — provide a reasonable model of local dependencies in phonology, including those created by iterated processes such as progressive vowel harmony. But strictly local grammars provably cannot regulate dependencies that involve two segments that can be arbitrarily far apart. An example of that is sibilant harmony in Samala, as discussed in Heinz (2015). In Samala, no word may contain both [s] and [ʃ], irrespective of how far the two are apart. Suppose one wanted to account for this with a strictly k-local grammar. This grammar would consist of all possible k-grams that contain both [s] and [ʃ]. But this grammar enforces sibilant harmony only as long as the two sibilants are separated by at most $k-2$ segments. If they are farther apart, they do not belong to the same k-gram and thus aren't regulated by the grammar. Long distance processes thus are beyond the capabilities of strictly local grammars.

The simplest way of accounting for such unbounded dependencies is to maintain the k-gram approach but change how k-grams are interpreted. In a strictly local grammar the bigram sʃ indicates that s must not be immediately followed by ʃ. But if we switch from *immediately follow* (the relation S) to *follow at any distance* (the relation $\prec$), k-grams enforce unbounded dependencies instead of local ones. Grammars where k-grams enforce unbounded dependencies are called *strictly k-piecewise*.

The first-order logic perspective reveals the close parallel between strictly local and strictly piecewise grammars. Both use the same template for formulas but use different relations. Just consider the strictly local and strictly piecewise formulas for the set {sʃ, ʃs}, which are almost identical.

$$\forall x_1, x_2 \Big[x_1 \; S \; x_2 \rightarrow \neg(\mathrm{s}(x_1) \wedge \int(x_2)) \wedge \neg(\int(x_1) \wedge \mathrm{s}(x_2)) \Big]$$

$$\forall x_1, x_2 \Big[x_1 \prec x_2 \rightarrow \neg(\mathrm{s}(x_1) \wedge \int(x_2)) \wedge \neg(\int(x_1) \wedge \mathrm{s}(x_2)) \Big]$$

While the strictly local interpretation of this set of bigrams only enforces harmony under adjacency, the strictly piecewise interpretation applies harmony across the entire word.

2.3 Extensions for Handling Primary Stress

Most attested dependencies that hold within a phonological word are either strictly local or strictly piecewise, but some require even more power, in particular suprasegmental dependencies. One example studied in detail in Heinz (2014) is primary word stress. It is a general property of phonological words that they have exactly one syllable carrying primary stress, meaning at least one and no more than one. But now consider a language where primary stress goes either on the first or the last syllable of the word depending on some other factors. This means that both stressed initial syllables and stressed final syllables constitute licit k-grams, so neither can be part of a strictly local grammar. Consequently, a strictly local grammar would fail to block strings with primary stress on both the first and the last syllable.

Strictly piecewise grammars improve on this as they can rule out words with two primary stresses. Assume that the vowel of a syllable carrying stress is denoted V́. Then the strictly 2-piecewise grammar {V́V́} prevents a stressed vowel from being followed by another stressed vowel, and thus no word can contain more than one primary stress. However, this captures only one half of the requirement that words have exactly one primary stress, it still remains to ensure that every word has at least one primary stress. This, strictly piecewise grammars are not capable of. One might expect that the 4-gram ⋊VV⋉ could rule out words without primary stress on the first or last syllable, but since there is no requirement for the vowels to be adjacent to the word edges, this 4-gram would actually rule out any word with two unstressed vowels. With the formalisms introduced so far, then, it is impossible to enforce the presence of at least one primary stress.

There are three classes in the subregular hierarchy that can fix this shortcoming, at least under certain assumptions. All three enrich strictly local grammars with a particular ability.

1. *strictly threshold testable*: counting
2. *locally testable*: implications between k-grams
3. *tier-based strictly local*: projection of phonological tiers

The most brute-force solution to the primary stress puzzle is to start out with a strictly k-local grammar and assign each k-gram an integer. If these integers encode the maximum number of times an item may occur, one obtains a *strictly* *threshold testable grammar*. These grammars were studied in detail in Ruiz et al. (1998), but they are of little use here. Instead, we want integers to encode a k-gram's minimum number of occurrences. I call these grammars (which have not been studied in a formal setting yet) *strictly co-threshold testable*. The presence of at least one primary stress is then guaranteed by the strictly co-threshold testable grammar $\left\{\left\langle \acute{\mathrm{V}}, 1\right\rangle\right\}$, which generates only those strings that contain at least one vowel with primary stress. This counting-based solution does not work, however, if we wish to replace V by the actual vowels of the language because an expanded grammar like $\{\langle \acute{\mathrm{a}}, 1\rangle, \langle \acute{\mathrm{u}}, 1\rangle\}$ requires the presence of both a stressed a and a stressed u. So the counting approach works only at a high level of abstraction.

The second option is to establish logical dependencies between k-grams that render the presence of one contingent on the absence of another. In the case at hand, we could posit that a string starts with a stressed syllable if and only if it does not end with a stressed syllable. In terms of first-order logic:

$$\exists x_1, x_2[x_1 \; S \; x_2 \wedge \rtimes(x_1) \wedge \acute{\mathrm{V}}(x_2)] \longleftrightarrow \neg\exists x_1, x_2[x_1 \; S \; x_2 \wedge \acute{\mathrm{V}}(x_1) \wedge \ltimes(x_2)]$$

Note that with this approach we can safely replace $\acute{\mathrm{V}}$ by the actual vowels of the language using logical disjunction.

$$\exists x_1, x_2[x_1 \; S \; x_2 \wedge \rtimes(x_1) \wedge (\acute{\mathrm{a}}(x_2) \vee \acute{\mathrm{u}}(x_2))] \longleftrightarrow \neg\exists x_1, x_2[x_1 \; S \; x_2 \wedge (\acute{\mathrm{a}}(x_1) \vee \acute{\mathrm{u}}(x_1)) \wedge \ltimes(x_2)]$$

Strictly local grammars that are supplemented with requirements of this form are called *locally testable grammars*.

The last option is the linguistically most natural: provide additional structure in the form of tiers. When determining whether a word contains an illicit sequence, a *tier-based strictly local grammar* (Heinz et al. 2011) ignores all segments whose labels do not belong to a predefined set. For primary stress assignment, we project a primary stress tier that contains only the word edges and any vowels carrying primary stress. The illicit bigrams over this stress tier are $\rtimes\ltimes$ (no primary stress) and $\acute{\mathrm{V}}\acute{\mathrm{V}}$ (more than one primary stress). Again we can safely replace V by all vowels of the language. Just like the locally testable grammars, then, tier-based strictly local grammars successfully enforce both the lower and the upper bound for primary stress and do not depend on an abstracted alphabet to do so.

We now have a variety of subregular classes, and we have seen that strictly local and strictly piecewise grammars are not sufficient to handle all aspects of natural language phonotactics. This raises the question which ones of the extensions introduced here can be considered good models of phonology, which is addressed next.

2.4 Empirical Viability of Subregular Classes

The previous section presented strictly (co-)threshold testable grammar, locally testable grammars, and tier-based strictly local grammars as conceivable enhancements of strictly local grammars. I will now argue on both conceptual and empirical grounds that among those three, only tier-based strictly local grammars are linguistically natural.

Conceptually, phonological tiers have long been entertained in the literature. The tiers of tier-based strictly local grammars are highly impoverished in that they do not accommodate autosegmental notions such as spreading or delinking, but they are still much more in line with phonological research than counting or k-gram interdependencies. The latter two also give rise to various kinds of overgeneration.

Strictly co-threshold testable grammars with their counting mechanism can put lower bounds on anything that can be expressed via a k-gram, be it primary stress or very elaborate consonant clusters. For example, one could easily write a small strictly co-threshold testable grammar that requires every word to contain at least two instances of the cluster [mpl]. This is impossible in a tier-based strictly local grammar: even if we project a tier that contains all segments labeled [m], [p] and [l] and require the tier to be sufficiently long, that still does not guarantee that the word contains two instances of [mpl]. The reason for this is that adjacency on the projected tier does not imply adjacency in the word. That is to say, the words [amplampla] and [amapalamapala] project exactly the same tier, so a tier-based strictly local grammar cannot block the latter without incorrectly blocking the former, too. The toy example shows that for strictly (co-)threshold testable grammars, any counting dependency is easy to handle, irrespective of whether it is phonologically natural. Tier-based strictly local grammars, on the other hand, resemble natural language phonology in that they can do some counting, as we learned during the discussion of primary stress, but only in select cases.

Locally testable grammars face a similar overgeneration issue. Loosely speaking, any boolean combination of k-grams is expressible in locally testable grammars. This includes ludicrous requirements such as “if the word contains [sl] or it contains [ls] but not [ma], then it ends in a vowel only if it starts with the same vowel and no instance of this vowel in the word is followed by [x]”. Natural languages simply do not impose dependencies of this kind. Even highly elaborate patterns can usually be explained in terms of the interaction of several very simple principles. Needless to say, tier-based strictly local grammars cannot enforce k-gram interdependencies of this kind.

The advantages of tier-based strictly local grammars emerge even more clearly once we discard an assumption that is commonly made in the subregular study of phonology: that the application domain of phonological rules is automatically restricted to a single phonological word. This clearly is an oversimplification since there are numerous phonological processes that apply across word boundaries. Once the domain is extended from single words to entire sentences, though, the highly unnatural classes of strictly (co-)threshold testable grammars and locally testable grammars also become empirically insufficient, whereas tier-based strictly local grammars are completely unaffected by this change.

Let us look at the effects of moving from words to sentences in greater detail. Assume that grammars now have to regulate the well-formedness of entire sentences instead of just individual words. More formally, the strings under consideration are now of the form $\rtimes\$w_1\$w_2\$\ldots\$w_n\$\ltimes$, where each w_i is some word, \$ indicates word edges, and $\rtimes$ and $\ltimes$ indicate the edges of the whole string as before.

Recall that the strictly co-threshold testable grammar required the presence of at least one vowel with primary stress. While this had the desired effect as long as the grammar described the shape of individual words, it clearly fails for sentences as it now ensures only the presence of at least one primary stress for the whole sentence. Increasing the minimum bound does not help either, since it depends on the number of words in the sentence and thus cannot be fixed in advance. Similarly, the locally testable solution that word-initial primary stress appears if and only if word-final primary stress is not present breaks down because some words in the sentence may have the former and some the latter — no language in the world requires all words in a sentence to display the same kind of primary stress pattern.

The tier-based strictly local solution, on the other hand, blocked configurations where the word edges are adjacent in the stress tier or separated by more than one segment — this solution works just as well over sentences. The move from words to sentences thus provides further evidence that only the tier-based strictly local grammars generalize the strictly local grammars in the right way.

It is also noteworthy that tier-based strictly local grammar handle long-distance dependencies better than the strictly piecewise grammars if one takes the step from words to sentences. In Sec. 2.2, sibilant harmony in Samala was captured via a strictly 2-piecewise grammar consisting of the bigrams sʃ and ʃs, which blocks any word that contains both s and an ʃ. But this solution fails with multi-word sequences. If word w_1 contains s and word w_2 contains ʃ, then the sentence $\rtimes\$w_1\$w_2\$\ltimes$ is ruled out even though it does not violate sibilant harmony. However, there is a tier-based account that works equally well for words and sentences. All we need to do is to project a sibilant tier that contains all instances of s, ʃ, and the word edge marker $. The illicit strings, then, are exactly those that contain the bigrams sʃ or ʃs. Since words are still separated by $ on the tier, these bigrams can only apply to sibilants that belong to the same word. Even for long-distance processes, then, tier-based strictly local grammars seem to provide a better fit.

The discussion so far suggests that tier-based strictly local grammars provide the best subregular model of natural language phonotactics. They are linguistically natural, capture a wide range of phenomena, do not display egregious overgeneration, and generalize well from words to strings. Unfortunately, though, there are still some aspects of natural language phonology that are beyond their reach. These are instances of so-called *circumambient* patterns (Jardine 2015), where a segment must take a specific shape whenever it occurs within an interval spanned by two other segments.

A well known example of a circumambient pattern is tone plateauing: a low tone becomes high if there is a high tone somewhere to its left and another high tone somewhere to its right. Over words, this is a strictly piecewise dependency as all illicit patterns are correctly ruled out by the single trigram HLH. Yet over sentences this account once again falters because the high tones must belong to the same word to trigger plateauing, something that the strictly piecewise grammar cannot keep track of. In contrast to sibilant harmony, however, tone plateauing cannot be regulated with tier-based strictly local grammars, either. Clearly such a grammar must use a tone tier that accommodates all segments with high tone or low tone. But since there is no upper bound on the number of low tones that may occur between two high tones, the high tones can still be arbitrarily far apart on this tier. So tier projection does not render tone plateauing strictly local, and consequently it cannot be regulated by a tier-based strictly local grammar.

The existence of circumambient patterns presents quite a conundrum. On the one hand, they can be handled by strictly piecewise grammars over words, suggesting that they are comparatively simple. At the same time, they exceed the capabilities of tier-based strictly local grammars, which otherwise exhibit many appealing properties. Ideally, it should be possible to unify the two formalisms into a single solution that can also handle circumambient patterns over sentences but does not overgenerate in the way that strictly threshold testable and locally testable grammars do. In the next section I show that this is indeed possible. Rather than strictly local grammars, I take strictly piecewise grammars as my vantage point and generalize them by adding a mechanism for creating locality domains.

3 Domain-Based Strictly Piecewise Grammars

3.1 Definition and Examples

The reason that tier-based strictly local formalisms can be successfully used at the word-level as well as the sentence-level is that they already come with a built-in locality domain. Since they never apply to more than k consecutive segments, they are aware of word boundaries and may take these into account in determining the well-formedness of the input. The strictly piecewise grammars, on the other hand, gain their ability to regulate segments that are arbitrarily far apart at the expense of any awareness of the material that occurs between these segments, including any word boundaries. As a result, strictly piecewise grammars cannot distinguish between well-formedness at the word-level and well-formedness at the sentence-level — a word-internal pattern like tone plateauing will be indiscriminately enforced across all words in a sentence. The obvious solution is to equip strictly piecewise grammars with a specification of the locality domain within which they may apply. Somewhat surprisingly, this doesn't just patch a hole in the formalism but provides a novel, unified perspective of the subregular hierarchy in terms of locality domains.

The unifying power of domain restrictions is readily apparent once the idea is carefully spelled out in formal terms.

Definition 1 (Domain-Based SP)**.** *A* domain-based strictly k-piecewise *grammar is a tuple $G := \langle P, D\rangle$, where P is a strictly k-piecewise grammar and D is a formula of first-order logic, which we call the* domain requirement of P *or simply the* domain. *A string s is generated by G iff the first-order formula below is true in s:*

$$\forall x_1, \dots, x_k [D \rightarrow \bigwedge_{p \in P} \neg \phi_p]$$

Here ϕ_p denotes the first-order equivalent of the labeling information inherent in the strictly piecewise k-gram p (cf. Sec. 2.1). For instance, the bigram *ab* corresponds to the formula $a(x_1) \wedge b(x_2)$. The big *and* operator acts as a shorthand for multiple conjunctions. Assuming that the grammar contains only the bigrams ⋊*a*, *aa*, and *a*⋉, the expanded formula would be:

$$\forall x_1, \dots, x_k [D \rightarrow \neg(\rtimes(x_1) \wedge a(x_2)) \wedge \neg(a(x_1) \wedge a(x_2)) \wedge \neg(a(x_1) \wedge \ltimes(x_2))$$

The overall structure of the formula ensures that the conditions enforced by the strictly piecewise grammar need to be met by nodes $x_1, \dots, x_k$ only if they jointly satisfy the domain requirement D.

Tone plateauing is now easily restricted to the word level using a small domain-based strictly 3-piecewise grammar $G := \langle P, W\rangle$. As in Sec. 2.4, P only contains the trigram *HLH* (or all the compiled out variants of this pattern over a phone-based alphabet). Assuming as before that word edges are explicitly marked by \$, the word domain restriction W is as follows:

$$W := \exists l, r \Big[l \prec x_1 \wedge \bigwedge_{1 \leq i < 3} x_i \prec x_{i+1} \wedge x_3 \prec r \wedge \$(l) \wedge \$(r) \wedge \neg \exists z [\$(z) \wedge l \prec z \wedge z \prec r] \Big]$$

This formula may look prohibitive, but enforces a very simple property: x_1, x_2, and x_3 must occur between word edge markers l and r that are not separated by another word edge marker z. In other words, x_1, x_2, and x_3 must belong to the same word. Only if this condition is met are the three nodes subjected to the requirement enforced by the trigram *HLH*. The full first-order sentence for G is given here to illustrate how exactly the different pieces of the grammar are combined into a single formula:

$$\begin{aligned}&\forall x_1, x_2, x_3\Big[\\ &\exists l, r\Big[l \prec x_1 \wedge \bigwedge_{1\leq i<3} x_i \prec x_{i+1} \wedge x_3 \prec r \wedge \$(l) \wedge \$(r) \wedge \neg\exists z[\$(z) \wedge l \prec z \wedge z \prec r]\Big] \rightarrow\\ &\qquad\qquad\qquad\qquad\qquad\qquad \neg\Big(H(x_1) \wedge L(x_2) \wedge H(x_3)\Big)\Big]\end{aligned}$$

Note that tone plateauing can be expressed by an even simpler grammar that contains only the unigram L. This requires us to change the domain from words to word-internal strings that occur between two high tones.

$$HH := \exists l, r\Big[l \prec x_1 \wedge x_1 \prec r \wedge H(l) \wedge H(r) \wedge \neg\exists z\Big[\big(H(z) \vee \$(z)\big) \wedge l \prec z \wedge z \prec r\Big]\Big]$$

The domain restriction HH differs only marginally from W. The left and right edge are now required to be high tones instead of word edges, and no segment between the two edges may itself be a high tone or a word edge. The former ensures that we pick out the shortest intervals between high tones, the latter that we do not apply tone plateauing across two words. With HH instead of W we thus obtain a different description of tone plateauing that trades a slightly more complicated domain restrictor for a simpler strictly piecewise grammar.

$$\forall x_1\Big[\exists l, r\Big[l \prec x_1 \wedge x_1 \prec r \wedge H(l) \wedge H(r) \wedge \neg\exists z\Big[\big(H(z) \vee \$(z)\big) \wedge l \prec z \wedge z \prec r\Big]\Big] \rightarrow \neg L(x_1)\Big]$$

3.2 Relation to (Tier-Based) Strict Locality

Domain restrictions are a powerful tool. Not only do they make it straightforward to limit strictly piecewise dependencies to specific (morpho-)phonological domains such as the word, the syllable, or specific morphemes, they also allow us to consider strictly local and tier-based strictly local dependencies as special cases of strictly piecewise ones. Intuitively, a strictly k-local grammar is a strictly k-piecewise grammar where the locality domain is restricted to k consecutive segments. A tier-based strictly k-local grammar just ignores non-projecting segments for the computation of this locality domain. Assuming that all phonological dependencies belong to one of these three classes, they all turn out to be the result of restricting general precedence requirements to specific locality domains.

Let us consider the domain restrictions for strictly local and tier-based strictly local dependencies in greater detail, and compare them to the domain restrictions W and HH above. A strictly k-local grammar has the domain restriction SL_k to enforce that each x_i immediately precedes x_{i+1}.

$$SL_k := \bigwedge_{1 \leq i < k} \Big(x_i \prec x_{i+1} \land \neg\exists z [x_i \prec z \land z \prec x_{i+1}] \Big)$$

A tier-based strictly k-local grammar uses the same domain restriction template, but filters out nodes that aren't labeled with a symbol that projects to tier T.

$$TSL_k := \bigwedge_{1 \leq i < k} \Bigg(\Big(\bigvee_{t \in T} t(x_i) \Big) \land \Big(\bigvee_{t \in T} t(x_i+1) \Big) \land x_i \prec x_{i+1} \land \neg\exists z \Bigg[\Big(\bigvee_{t \in T} t(z) \Big) \land x_i \prec z \land z \prec x_{i+1} \Bigg] \Bigg)$$

The domain restrictions exhibit certain similarities to W and HH from the previous subsection. In all three cases we have a conjunct $\neg\exists z[\phi]$ that blocks the existence of some node z occurring between two nodes. In addition, W, HH and TSL_k only mind the presence of z if it carries a specific label, indicated by disjunctions such as $\bigvee_{t \in T} t(z)$ in TSL_k. The first half of the formulas also follows a common template where nodes are linearly ordered with respect to each other and possibly required to carry specific labels.

There is a minor difference, though, in how the linear order requirements are enforced. Both W and HH pick out edges l and r with specific labels, require all x_i to occur between l and r, ensure that each x_i precedes x_{i+1}, and permit no z between l and r. In SL_k and TSL_k, on the other hand, the domain lacks l and r, and no z may occur between any x_i and x_{i+1}. However, it is possible to redefine SL_k and TSL_k in a fashion that resembles the format of W and HH more closely.

$$SL'_k := \exists l, r \Bigg[\bigwedge_{1 \leq i < k} x_i \prec x_{i+1} \land \neg\exists z \Bigg[x_1 \prec z \land z \prec x_k \land \bigwedge_{1 < i < k} z \neq x_i \Bigg] \Bigg]$$

$$TSL'_k := \exists l, r \Bigg[\bigwedge_{1 \leq i < k} x_i \prec x_{i+1} \land \bigwedge_{1 \leq i \leq k} \Big(\bigvee_{t \in T} t(x_i) \Big) \land \neg\exists z \Bigg[x_1 \prec z \land z \prec x_k \land \bigwedge_{1 < i < k} z \neq x_i \land \bigvee_{t \in T} t(z) \Bigg] \Bigg]$$

The use of $\exists l, r$ only serves to highlight the parallels between all four domain restrictions. Since l and r are never referenced in the rest of the formula, this is an instance of vacuous quantification that does not affect the truth conditions of the formula.

3.3 Unification via Intervals

Given the alternative definitions SL'_k and TSL'_k, all four domain restrictions from the previous subsection can be understood as instantiations of a specific domain template $\mathscr{D}$:

$$\begin{aligned}\mathscr{D} := \exists l, r[&\mathit{Precedence}\ \langle \wedge \mathit{Labels}\rangle\ \wedge \\ &\neg\exists z[\mathit{LeftEdge} \prec z \wedge z \prec \mathit{RightEdge}\ \langle \wedge z\text{-}\mathit{Distinctness}\rangle\ \langle \wedge z\text{-}\mathit{Labels}\rangle]]\end{aligned}$$

I use pointy brackets here to indicate parts that may be omitted.

Each piece of the template has a specific form. The first part, *Precedence*, is a conjunction of precedence statements defining a linear order over all x_i and possibly including l and r as the first and last element, respectively.

$$\mathit{Precedence} := \bigwedge_{1 \leq i < k} x_i \prec x_{i+1}\ \langle \wedge l \prec x_1 \rangle\ \langle \wedge x_k \prec r \rangle$$

This is followed by *Labels*, a conjunction of disjunctions that regulate what labels l, r and each x_i may display.

$$\mathit{Labels} := \bigwedge_{1 \leq i \leq k} \bigvee_{\sigma \in \Sigma_i} \sigma(x_i)\ \langle \wedge \bigvee_{\sigma \in \Sigma_l} \sigma(l) \rangle\ \langle \wedge \bigvee_{\sigma \in \Sigma_r} \sigma(r) \rangle$$

As a notational shorthand, Σ_n is used to denote the set of licit labels for node n.

Within the existentially quantified part of the template, the placeholders *LeftEdge* and *RightEdge* are replaced by whatever nodes mark the left and right edge of the domain. This will be l or x_1, and r or x_k, depending on whether l and r are used. The z-*Distinctness* criterion states that z is distinct from all other nodes that occur between the edges of the domain.

$$z\text{-}\mathit{Distinctness} := \bigwedge_{1 < i < k} z \neq x_i\ \langle \wedge z \neq x_1 \rangle\ \langle \wedge z \neq x_k \rangle$$

Finally, z-*Labels* is a disjunction listing the labels for z.

$$z\text{-}\mathit{Labels} := \bigvee_{\sigma \in \Sigma_z} \sigma(z)$$

Table 1 summarizes how the four domain restrictions discussed so far are instantiations of $\mathscr{D}$.

The template shows that domain-based strictly piecewise grammars unify

	W	HH	SL'_k	TSL'_k
Precedence	$l \prec x_1 \wedge \bigwedge_{1 \le i < 3} x_i \prec x_{i+1} \wedge x_3 \prec r$	$l \prec x_1 \wedge x_1 \prec r$	$\bigwedge_{1 \le i < k} x_i \prec x_{i+1}$	$\bigwedge_{1 \le i < k} x_i \prec x_{i+1}$
Labels	$\$(l) \wedge \$(r)$	$H(l) \wedge H(r)$	—	$\bigwedge_{1 \le i \le k} \bigvee_{t \in T} t(x_i)$
LeftEdge	l	l	x_1	x_1
RightEdge	r	r	x_k	x_k
z-Distinctness	—	—	$\bigwedge_{1 < i < k} z \neq x_i$	$\bigwedge_{1 < i < k} z \neq x_i$
z-Labels	$\$(z)$	$H(z) \vee \$(z)$	—	$\bigvee_{t \in T} t(z)$

Table 1: Domain restrictions as instantiations of the general domain template

various subregular classes in an interesting way. Strictly local and tier-based strictly local dependencies turn out to be instances of the kind of domain restriction that is also needed to limit long-distance processes to the word domain. Let us refer to domain-based strictly piecewise grammars whose domain restriction follows the general domain template as *interval-based strictly piecewise grammars.*

Definition 2 (Interval-Based SP)**.** *A domain-based strictly piecewise grammar $G := \langle D, P \rangle$ is* interval-based *iff D is of the form indicated by the template $\mathscr{D}$.*

The interval-based strictly piecewise grammars offer a reasonable model of phonotactic dependencies. They can handle the same data as strictly local and tier-based strictly local languages, including all local processes, long-distance dissimilation, and primary stress. But they improve on these by also including all strictly piecewise dependencies (in the case where the domain restrictor picks out the entire string) as well as strictly piecewise dependencies limited to unboundedly large subdomains such as words or morphologically complex subparts of words. The latter is what allows them to also handle tone plateauing over entire sentences, in contrast to tier-based strictly local grammars. In fact, circumabmient patterns such as tone plateauing are expected to exist since they merely co-opt mechanisms of the grammar that are independently needed to restrict strictly piecewise dependencies to the word domain.

In addition, interval-based strictly piecewise grammars also offer a natural account for blocking effects. Suppose that tone plateauing is blocked whenever the span between the two high tones contains some segment labeled B. This requires just a minor alteration in the definition of the *HH* domain — z may not be labeled B either.

$$HH' := \exists l, r\Big[l \prec x_1 \wedge x_1 \prec r \wedge H(l) \wedge H(r) \wedge \neg \exists z\big[\big(H(z) \vee \$(z) \vee B(z)\big) \wedge l \prec z \wedge z \prec r\big]\Big]$$

Overall, then, interval based strictly piecewise grammars offer an insightful and empirically viable unification of all phonologically interesting subregular classes posited so far, a unification that is grounded in one of the most natural of linguistic concepts: locality domains. In formal terms, interval-based strictly piecewise languages constitute a new subregular language class that is more powerful then the classes it subsumes.

Theorem 1. *The class of interval-based strictly piecewise languages is a proper superclass of the union of the (tier-based) strictly local languages and the strictly piecewise languages.*

The importance and restrictive power of natural locality domains is confirmed in the next section, where I show that domain-based strictly piecewise grammars are much more powerful than their interval-based subtype. But this additonal power is unweclome as it endows the grammars with the ability to count and establish elaborate interdepenencies between k-grams, both of which were argued against in Sec. 2.4. In addition, the limitation to interval domains ensures learnability, whereas the full class of domain-based strictly piecewise grammars is not learnable.

3.4 Generative Capacity

Interval-based strictly piecewise grammars form a severely restricted subclass of the domain-based strictly piecewise grammars. This may not be readily apparent since from a linguistic perspective the domain restriction template forced upon interval-based strictly piecewise grammars can still capture pretty much any phonologically relevant locality domain. The reason for the marked disparity between interval-based and domain-based strictly piecewise grammars thus is not due to the weakness of the former but rather the excessive power of the latter. It turns out that the class of domain-based strictly piecewise languages is exactly the class of star-free languages, which resides at the very top of the subregular hierarchy (with only the class of regular languages above it; McNaughton and Pappert 1971).

Theorem 2. *Every domain-based strictly piecewise language is star-free, and* vice versa.

The proof of this equivalence exploits the permissive definition of domain restrictions. Recall from Sec. 3.1 that a domain-based strictly piecewise grammar only requires domain restrictions to be first-order formulas. The domain restriction D is then inserted into a general template $\forall x_1, \ldots, x_k[D \rightarrow \bigwedge_{p \in P} \neg\phi_p]$. But this procedure is so general that D can be used to impose arbitrary constraints as long as they are first-order definable. And the star-free languages are exactly those that can be defined by arbitrary first-order definable constraints.

Proof. Let L be some star-free language. Then there is some first-order formula ϕ_L without free variables such that L is the set of string models of ϕ_L. Now it holds for any given string w that it satisfies ϕ_L iff it satisfies $\forall x_1[\neg\phi_L \rightarrow \bot]$, where $\bot$ is a shorthand for some formula that is always false. This bi-implication holds because I) ϕ_L contains no free variables that can be bound by $\forall x_1$, and II) by the definition of implication, $a = 1$ iff $\neg a \rightarrow 0 = 1$. Observe in addition that the strictly 1-piecewise grammar containing only the unigram $\rtimes$ generates the empty language, which means that no string satisfies $\forall x_1[\neg \rtimes (x_1)]$. Putting all of this together, we conclude that L is generated by the domain-based strictly 1-piecewise grammar $G := \langle \neg\phi_L, \{\rtimes\} \rangle$. Since L was arbitrary, every star-free language is domain-based strictly 1-piecewise.

In the other direction, the inclusion of all domain-based strictly piecewise languages in the class of star-free languages is implied by the fact that domain-based strictly piecewise languages are first-order definable. □

The proof does not carry over to interval-based strictly piecewise languages because the first-order formula for a given star-free language is not guaranteed to comply with the domain restriction template of this subclass. Rather than show this directly, I give examples of strictly (co-)threshold testable and locally testable languages that are not interval-based strictly piecewise. Since the former are all star-free, the interval-based strictly piecewise languages must be a proper subclass of the star-free languages.

Let us first see why interval-based strictly piecewise grammars do not have the same counting power as strictly (co-)threshold testable grammars. Recall from Sec. 2.3 that a strictly threshold testable grammar is a strictly local grammar where each k-gram is associated with a natural number n that indicates the maximum number of times the k-gram may occur in the string. In a strictly co-threshold testable grammar, on the other hand, n indicates the minimum number of times the k-gram must occur.

Now suppose language L is such that every well-formed word contains at most one instance of VN, a vowel immediately followed by a nasal. This language is strictly threshold testable, but it is not interval-based strictly piecewise. In an interval-based strictly piecewise grammar, the domain restriction must define a continuous interval within which the k-grams of the strictly piecewise grammar apply. Assume that I is such an interval that spans at least four nodes $x_1, \ldots, x_4$ such that $V(x_1)$, $N(x_2)$, $V(x_3)$, and $N(x_4)$. We can block this configuration with the 4-gram VNVN. While this ensures that no string contains two instances of VN, it blocks many other strings that should not be ruled out: any string where I contains two vowels v_1 and v_2 and two nasals n_1 and n_2 with $v_1 \prec n_1 \prec v_1 \prec n_2$ is prohibited, even if none of the four segments are adjacent. In order to enforce adjacency, one has to instantiate the domain restriction with the z-*Distinctness* parameter and restrict the labels of z to vowels and nasals. But such a grammar undergenerates since it fails to block at least some strings of the form VNxVN, where x is a sequence of one or more vowels. The reader is invited to play with some of the other parameters of the template $\mathscr{D}$, but the final answer remains the same: there is no interval-based strictly piecewise grammar for L.[1]

A similar argument can be used to show that the variant of L that contains at least two instances of VN is not interval-based strictly piecewise. Some strictly (co-)threshold testable patterns are interval-based strictly piecewise, though, by virtue of being tier-based strictly local. Primary stress assignment is the empirically most noteworthy among those.

Lemma 1. *The class of interval-based strictly piecewise languages is incomparable with both the class of strictly locally testable languages and the class of co-strictly locally testable languages.*

This is a welcome result as it captures the old adage that "languages do not count". Where counting effects seem to obtain, as with primary stress assignment, they have to be so simple that they can be captured via purely structural means, e.g. via tier projection.

Another example language establishes that interval-based strictly piecewise grammars and locally testable grammars define distinct language classes. Consider a language where every word must start with a consonant or end with a

[1]A workable solution would have to use an interleaved instantiation of z-*Distinctness* and z-*Labels*:

$$\neg\exists z[(x_1 \prec z \wedge z \prec x_2) \vee (x_3 \prec z \wedge z \prec x_4)] \wedge \neg\exists z[l \prec z \wedge z \prec r \wedge \bigwedge_{1 \leq i \leq 4} z \neq x_i \wedge (V(z) \vee N(z))]$$

But this is not permitted by the template $\mathscr{D}$.

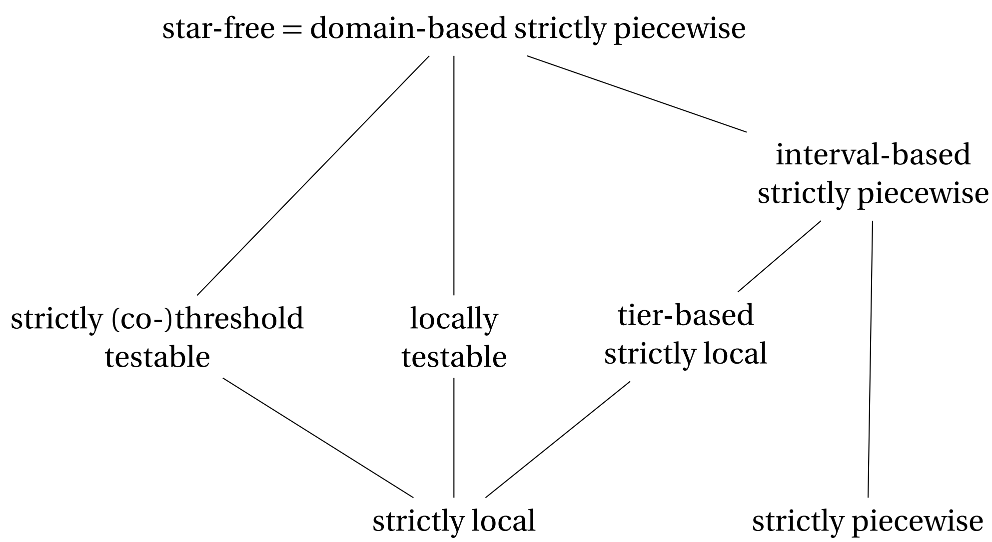

Figure 1: Subsumption relations between the subregular classes discussed in this paper

consonant, though it can both start and end with a consonant. The only illicit strings are those that start and end with vowels. The locally testable grammar capturing this behavior corresponds to the constraint "it is not the case that \$V and V\$ are bigrams of the string". In order to regulate this behavior with an interval-based strictly local grammar, the domain restriction has to include the word edges, wherefore the domain — which must be continuous — spans the entire word. The pattern we want to block is \$$xy$\$, where x and y are the first and last vowel of the word, respectively. But since the domain spans the entire word, x and y can be identified with any vowels, and consequently the grammar would block any word containing two vowels. As before, the problem is that the condition mixes strictly local information (\$ and x are adjacent, and so are y and \$) with locally unbounded information (x precedes y but the two need not be adjacent). This kind of mixing is not possible with interval-based strictly piecewise grammars.

Lemma 2. *The class of interval-based strictly piecewise languages is incomparable with the class of locally testable languages.*

Corollary 1. *The interval-based strictly piecewise grammars generate a proper subclass of the domain-based strictly piecewise grammars.*

Figure 1 depicts the subsumption relations between the various subregular classes.

3.5 Learnability

The large gap in expressivity between domain-based and interval-based strictly piecewise languages not only shows the restrictive nature of locality domains in phonology, it is also an essential factor for learnability. It is a well-known fact that the star-free language are not learnable in the limit from positive text (a corollary of them properly subsuming the class of finite languages, cf. Gold 1967). Consequently, the expressively equivalent class of domain-based strictly piecewise languages cannot be learned in the limit from positive text, either. However, the interval-based strictly k-piecewise languages are learnable for any fixed choice of k. This learnability claim is trivial once one realizes that there can be only finitely many distinct languages in this class for any choice of k. First, there are only finitely many strictly k-piecewise grammars because there are only finitely many distinct sets of k-grams over the alphabet. Second, only finitely many distinct domain restrictions can be obtained from the template $\mathscr{D}$ if the number of variables x_i must be at most k. If there are m strictly k-piecewise grammars and n instantiations of $\mathscr{D}$ to choose from, the number of interval-based strictly k-piecewise grammars languages is $m \cdot n$, and hence the number of languages is at most $m \cdot n$. Crucially, the finiteness of a language class implies its learnability, as was already proved in Gold (1967).

Theorem 3. *The class of interval-based strictly k-piecewise languages is learnable in the limit from positive text.*

This confirms the long-held belief that UG-restrictions on locality domains aid learnability: a child that comes already equipped with the assumption that dependencies involve k elements and may only be relativized to locality domains of a specific shape can successfully learn any language in the hypothesis space. Lifting the restriction on what locality domains may look like destroys learnability as even the restriction to $k = 1$ still allows domain-based strictly piecewise grammars to generate any star-free language.

Conclusion

While a lot of progress has been made during the last decade when it comes to mapping out the areas of the subregular hierarchy occupied by phenomena of natural language phonology, the mechanism for creating the domains within which phonological patterns must be regulated has been neglected so far. The surprising result of this paper is that taking these domains into account simplifies the overall picture rather than complicating it. The full range of subregular classes can now be understood to arise from strictly piecewise grammars amplified with domain restrictions. Even more importantly, limiting these domain restrictions to linguistically natural intervals has a number of welcome effects:

1. Overgeneration is curbed by ruling out unnatural dependencies involving counting or elaborate conditionals.
2. Undergeneration is avoided since all strictly piecewise, strictly local, and tier-based strictly local languages can still be generated.
3. Circumambient patterns such as tone plateauing arise in a natural fashion from the ability to define domain restrictions.
4. Blocking effects in circumambient patterns can be accommodated, too.
5. The internal structure of the language space is still sufficiently limited to allow for learning in the limit from positive text.

Since the domains within which phonological processes apply have been studied by linguists for many years, there are many ideas in the literature that one can draw from in order to generalize or further refine these findings. Careful exploration of the formal ramifications of such modifications should yield an even tighter characterization of natural language phonotactics and might also prove fruitful in other areas.

For example, Graf and Heinz (2015) argue that syntax can be regarded as tier-based strictly local at a sufficiently high level of abstraction. This finding relies on certain simplifying assumptions about movement, though, that are slightly at odds with the current thinking in Minimalist syntax. Interval-based strictly piecewise grammars may offer enough headroom to capture syntactic dependencies in a more general setting. This paper thus constitutes but a first step towards a more thorough exploration of how locality domains can be elegantly incorporated into the study of subregular dependencies in language.

References

Chandlee, Jane. 2014. *Strictly local phonological processes.* Doctoral Dissertation, University of Delaware.

Gold, E. Mark. 1967. Language identification in the limit. *Information and Control* 447–474.

Graf, Thomas. 2010a. Comparing incomparable frameworks: A model theoretic approach to phonology. *University of Pennsylvania Working Papers in Linguistics* 16:Article 10. Available at: `http://repository.upenn.edu/pwpl/vol16/iss1/10`.

Graf, Thomas. 2010b. *Logics of phonological reasoning*. Master's thesis, University of California, Los Angeles.

Graf, Thomas, and Jeffrey Heinz. 2015. Commonality in disparity: The computational view of syntax and phonology. Slides of a talk given at GLOW 2015, April 18, Paris, France.

Heinz, Jeffrey. 2009. On the role of locality in learning stress patterns. *Phonology* 26:303–351.

Heinz, Jeffrey. 2010. Learning long-distance phonotactics. *Linguistic Inquiry* 41:623–661.

Heinz, Jeffrey. 2014. Culminativity times harmony equals unbounded stress. In *Word stress: Theoretical and typological issues*, ed. Harry van der Hulst, chapter 8. Cambridge, UK: Cambridge University Press.

Heinz, Jeffrey. 2015. The computational nature of phonological generalizations. Ms., University of Delaware.

Heinz, Jeffrey, Chetan Rawal, and Herbert G. Tanner. 2011. Tier-based strictly local constraints in phonology. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, 58–64.

Jardine, Adam. 2015. Computationally, tone is different. *Phonology* To appear.

Johnson, C. Douglas. 1972. *Formal aspects of phonological description*. The Hague: Mouton.

Kaplan, Ronald M., and Martin Kay. 1994. Regular models of phonological rule systems. *Computational Linguistics* 20:331–378.

McNaughton, Robert, and Seymour Pappert. 1971. *Counter-free automata*. Cambridge, MA: MIT Press.

Rogers, James, Jeffrey Heinz, Gil Bailey, Matt Edlefsen, Molly Vischer, David Wellcome, and Sean Wibel. 2010. On languages piecewise testable in the strict sense. In *The mathematics of language*, ed. Christan Ebert, Gerhard Jäger, and Jens Michaelis, volume 6149 of *Lecture Notes in Artificial Intelligence*, 255–265. Heidelberg: Springer.

Rogers, James, Jeffrey Heinz, Margaret Fero, Jeremy Hurst, Dakotah Lambert, and Sean Wibel. 2013. Cognitive and sub-regular complexity. In *Formal Grammar*, ed. Glyn Morrill and Mark-Jan Nederhof, volume 8036 of *Lecture Notes in Computer Science*, 90–108. Springer.

Ruiz, José, Salvador España, and Pedro García. 1998. Locally threshold testable languages in strict sense: Application to the inference problem. In *Grammatical inference: 4th international colloquium, ICGI-98 Ames, Iowa, USA, July 12–14, 1998 proceedings*, ed. Vasant Honavar and Giora Slutzki, 150–161. Berlin: Springer.