Tier-based Strictly Local Constraints for Phonology

Jeffrey Heinz, Chetan Rawal and Herbert G. Tanner
University of Delaware
heinz,rawal,btanner@udel.edu

Abstract

Beginning with Goldsmith (1976), the phonological tier has a long history in phonological theory to describe non-local phenomena. This paper defines a class of formal languages, the Tier-based Strictly Local languages, which begin to describe such phenomena. Then this class is located within the Subregular Hierarchy (McNaughton and Papert, 1971). It is found that these languages contain the Strictly Local languages, are star-free, are incomparable with other known sub-star-free classes, and have other interesting properties.

1 Introduction

The phonological tier is a level of representation where not all speech sounds are present. For example, the vowel tier of the Finnish word *päivää* 'Hello' is simply the vowels in order without the consonants: *äiää*.

Tiers were originally introduced to describe tone systems in languages (Goldsmith, 1976), and subsequently many variants of the theory were proposed (Clements, 1976; Vergnaud, 1977; McCarthy, 1979; Poser, 1982; Prince, 1984; Mester, 1988; Odden, 1994; Archangeli and Pulleyblank, 1994; Clements and Hume, 1995). Although these theories differ in their details, they each adopt the premise that representational levels exist which exclude certain speech sounds.

Computational work exists which incorporates and formalizes phonological tiers (Kornai, 1994; Bird, 1995; Eisner, 1997). There are also learning algorithms which employ them (Hayes and Wilson, 2008; Goldsmith and Riggle, to appear). However, there is no work of which the authors are aware that addresses the expressivity or properties of tier-based patterns in terms of formal language theory.

This paper begins to fill this gap by defining Tier-Based Strictly Local (TSL) languages, which generalize the Strictly Local languages (McNaughton and Papert, 1971). It is shown that TSL languages are necessarily star-free, but are incomparable with other known sub-star-free classes, and that natural groups of languages within the class are string extension learnable (Heinz, 2010b; Kasprzik and Kötzing, 2010). Implications and open questions for learnability and Optimality Theory are also discussed.

Section 2 reviews notation and key concepts. Section 3 reviews major subregular classes and their relationships. Section 4 defines the TSL languages, relates them to known subregular classes, and section 5 discusses the results. Section 6 concludes.

2 Preliminaries

We assume familiarity with set notation. A finite alphabet is denoted Σ. Let Σ^n, $\Sigma^{\leq n}$, Σ^* denote all sequences over this alphabet of length n, of length less than or equal to n, and of any finite length, respectively. The empty string is denoted λ and $|w|$ denotes the length of word w. For all strings w and all nonempty strings u, $|w|_u$ denotes the number of occurrences of u in w. For instance, $|aaaa|_{aa} = 3$. A language L is a subset of Σ^*. The concatenation of two languages $L_1L_2 = \{uv : u \in L_1 \text{ and } v \in L_2\}$. For $L \subseteq \Sigma^*$ and $\sigma \in \Sigma$, we often write $L\sigma$ instead of $L\{\sigma\}$.

We define generalized regular expressions (GREs) recursively. GREs include λ, $\emptyset$ and each letter of Σ. If R and S are GREs then RS, $R + S$, $R \times S$, $\overline{R}$, and R^* are also GREs. The language of a GRE is defined as follows.

Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics:shortpapers, pages 58–64,
Portland, Oregon, June 19-24, 2011.

$L(\emptyset) = \emptyset$. For all $\sigma \in \Sigma \cup \{\lambda\}$, $L(\sigma) = \{\sigma\}$. If R and S are regular expressions then $L(RS) = L(R)L(S)$, $L(R+S) = L(R) \cup L(S)$, and $L(R \times S) = L(R) \cap L(S)$. Also, $L(\overline{R}) = \Sigma^* - L(R)$ and $L(R^*) = L(R)^*$. For example, the GRE $\overline{\emptyset}$ denotes the language Σ^*.

A language is *regular* iff there is a GRE defining it. A language is *star-free* iff there is a GRE defining it which contains no instances of the Kleene star (*). It is well known that the star-free languages (1) are a proper subset of the regular languages, (2) are closed under Boolean operations, and (3) have multiple characterizations, including logical and algebraic ones (McNaughton and Papert, 1971).

String u is a *factor* of string w iff $\exists x, y \in \Sigma^*$ such that $w = xuy$. If also $|u| = k$ then u is a k-*factor* of w. For example, ab is a 2-factor of $aaabbb$. The function F_k maps words to the set of k-factors within them.

$$F_k(w) = \{u : u \text{ is a } k\text{-factor of } w\}$$

For example, $F_2(abc) = \{ab, bc\}$.

The domain F_k is generalized to languages $L \subseteq \Sigma^*$ in the usual way: $F_k(L) = \cup_{w \in L} F_k(w)$. We also consider the function which *counts* k-factors up to some threshold t.

$$\begin{aligned} F_{k,t}(w) = \quad & \{(u,n) : u \text{ is a } k\text{-factor of } w \text{ and} \\ & n = |w|_u \text{ iff } |w|_u < t \text{ else } n = t\} \end{aligned}$$

For example $F_{2,3}(aaaaab) = \{(aa, 3), (ab, 1)\}$.

A string $u = \sigma_1\sigma_2 \cdots \sigma_k$ is a *subsequence* of a string w iff $w \in \Sigma^*\sigma_1\Sigma^*\sigma_2\Sigma^* \cdots \Sigma^*\sigma_k\Sigma^*$. Since $|u| = k$ we also say u is a *k-subsequence* of w. For example, ab is a 2-subsequence of $caccccccccbcc$. By definition λ is a subsequence of every string in Σ^*. The function $P_{\leq k}$ maps words to the set of subsequences up to length k found in those words.

$$P_{\leq k}(w) = \{u \in \Sigma^{\leq k} : u \text{ is a subsequence of } w\}$$

For example $P_{\leq 2}(abc) = \{\lambda, a, b, c, ab, ac, bc\}$. As above, the domains of $F_{k,t}$ and $P_{\leq k}$ are extended to languages in the usual way.

3 Subregular Hierarchies

Several important subregular classes of languages have been identified and their inclusion relationships have been established (McNaughton and Papert, 1971; Simon, 1975; Rogers and Pullum, to appear; Rogers et al., 2010). Figure 1 summarizes those earlier results as well as the ones made in this paper. This section defines the Strictly Local (SL), Locally Threshold Testable (LTT) and Piecewise Testable (PT) classes. The Locally Testable (LT) languages and the Strictly Piecewise (SP) languages are discussed by Rogers and Pullum (to appear) and Rogers et al. (2010), respectively. Readers are referred to these papers for additional details on all of these classes. The Tier-based Strictly Local (TSL) class is defined in Section 4.

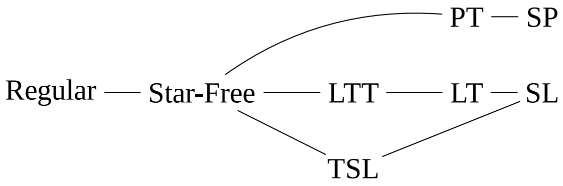

Figure 1: Proper inclusion relationships among subregular language classes (indicated from left to right). This paper establishes the TSL class and its place in the figure.

Definition 1 *A language* L *is* Strictly k-Local *iff there exists a finite set* $S \subseteq F_k(\rtimes\Sigma^*\ltimes)$ *such that*

$$L = \{w \in \Sigma^* : F_k(\rtimes w \ltimes) \subseteq S\}$$

The symbols $\rtimes$ and $\ltimes$ invoke left and right word boundaries, respectively. A language is said to be *Strictly Local* iff there is some k for which it is Strictly k-Local. For example, let $\Sigma = \{a, b, c\}$ and $L = aa^*(b+c)$. Then L is Strictly 2-Local because for $S = \{\rtimes a, ab, ac, aa, b\ltimes, c\ltimes\}$ and every $w \in L$, *every* 2-factor of $\rtimes w \ltimes$ belongs to S.

The elements of S can be thought of as the *permissible* k-factors and the elements in $F_k(\rtimes\Sigma^*\ltimes) - S$ are the *forbidden* k-factors. For example, bb and $\rtimes b$ are forbidden 2-factors for $L = aa^*(b+c)$.

More generally, any SL language L excludes exactly those words with any forbidden factors; i.e., L is the intersection of the complements of sets defined to be those words which *contain* a forbidden factor. Note the set of forbidden factors is finite. This provides another characterization of SL languages (given below in Theorem 1).

Formally, let the *container* of $w \in \rtimes\Sigma^*\ltimes$ be

$$C(w) = \{u \in \Sigma^* : w \text{ is a factor of } \rtimes u \ltimes\}$$

For example, $C(\rtimes a) = a\Sigma^*$. Then, by the immediately preceding argument, Theorem 1 is proven.

Theorem 1 *Consider any Strictly k-Local language L. Then there exists a finite set of forbidden factors $\bar{S} \subseteq F_k(\rtimes\Sigma^*\ltimes)$ such that $L = \cap_{w \in \bar{S}} \overline{C(w)}$.*

Definition 2 *A language L is* Locally t-Threshold k-Testable *iff $\exists t, k \in \mathbb{N}$ such that $\forall w, v \in \Sigma^*$, if $F_{k,t}(w) = F_{k,t}(v)$ then $w \in L \Leftrightarrow v \in L$.*

A language is Locally Threshold Testable iff there is some k and t for which it is Locally t-Threshold k-Testable.

Definition 3 *A language L is* Piecewise k-Testable *iff $\exists k \in \mathbb{N}$ such that $\forall w, v \in \Sigma^*$, if $P_{\leq k}(w) = P_{\leq k}(v)$ then $w \in L \Leftrightarrow v \in L$.*

A language is Piecewise Testable iff there is some k for which it is Piecewise k-Testable.

4 Tier-based Strictly Local Languages

This section provides the main results of this paper.

4.1 Definition

The definition of Tier-based Strictly Local languages is similar to the one for SL languages with the exception that forbidden k-factors only apply to elements on a tier $T \subseteq \Sigma$, all other symbols are ignored. In order to define the TSL languages, it is necessary to introduce an "erasing" function (sometimes called string projection), which erases symbols not on the tier.

$$E_T(\sigma_1 \cdots \sigma_n) = u_1 \cdots u_n$$

where $u_i = \sigma_i$ iff $\sigma_i \in T$ and $u_i = \lambda$ otherwise.

For example, if $\Sigma = \{a, b, c\}$ and $T = \{b, c\}$ then $E_T(aabaaacaaabaa) = bcb$. A string $u = \sigma_1 \cdots \sigma_n \in \rtimes T^* \ltimes$ is a *factor on tier T* of a string w iff u is a factor of $E_T(w)$.

Then the TSL languages are defined as follows.

Definition 4 *A language L is* Strictly k-Local on Tier T *iff there exists a tier $T \subseteq \Sigma$ and finite set $S \subseteq F_k(\rtimes T^* \ltimes)$ such that*

$$L = \{w \in \Sigma^* : F_k(\rtimes E_T(w) \ltimes) \subseteq S\}$$

Again, S represents the permissible k-factors on the tier T, and elements in $F_k(\rtimes T^* \ltimes) - S$ represent the forbidden k-factors on tier T. A language L is a *Tier-based Strictly Local* iff it is Strictly k-Local on Tier T for some $T \subseteq \Sigma$ and $k \in \mathbb{N}$.

To illustrate, let $\Sigma = \{a, b, c\}$, $T = \{b, c\}$, and $S = \{\rtimes b, \rtimes c, bc, cb, b\ltimes, c\ltimes\}$. Elements of S are the permissible k-factors on tier T. Elements of $F_2(\rtimes T^* \ltimes) - S = \{bb, cc\}$ are the forbidden factors on tier T. The language this describe includes words like *aabaaacaaabaa*, but excludes words like *aabaaabaaacaa* since *bb* is a forbidden 2-factor on tier T. This example captures the nature of long-distance dissimilation patterns found in phonology (Suzuki, 1998; Frisch et al., 2004; Heinz, 2010a). Let L_D stand for this particular dissimilatory language.

Like SL languages, TSL languages can also be characterized in terms of the forbidden factors. Let the *tier-based container* of $w \in \rtimes T^* \ltimes$ be $C_T(w) =$

$$\{u \in \Sigma^* : w \text{ is a factor on tier } T \text{ of } \rtimes u \ltimes\}$$

For example, $C_T(\rtimes b) = (\Sigma - T)^* b \Sigma^*$. In general if $w = \sigma_1 \cdots \sigma_n \in T^*$ then $C_T(w) =$

$$\Sigma^* \sigma_1 (\Sigma - T)^* \sigma_2 (\Sigma - T)^* \cdots (\Sigma - T)^* \sigma_n \Sigma^*$$

In the case where w begins (ends) with a word boundary symbol then the first (last) Σ^* in the previous GRE must be replaced with $(\Sigma - T)^*$.

Theorem 2 *For any $L \in TSL$, let T, k, S be the tier, length, and permissible factors, respectively, and $\bar{S}$ the forbidden factors. Then $L = \bigcap_{w \in \bar{S}} \overline{C_T(w)}$.*

Proof The structure of the proof is identical to the one for Theorem 1. □

4.2 Relations to other subregular classes

This section establishes that TSL languages properly include SL languages and are properly star-free. Theorem 3 shows SL languages are necessarily TSL. Theorems 4 and 5 show that TSL languages are not necessarily LTT nor PT, but Theorem 6 shows that TSL languages are necessarily star-free.

Theorem 3 *SL languages are TSL.*

Proof Inclusion follows immediately from the definitions by setting the tier $T = \Sigma$. □

The fact that TSL languages properly include SL ones follows from the next theorem.

Theorem 4 *TSL languages are not LTT.*

Proof It is sufficient to provide an example of a TSL language which is not LTT. Consider any threshold t and length k. Consider the TSL language L_D discussed in Section 4.1, and consider the words

$$w = a^k ba^k ba^k ca^k \text{ and } v = a^k ba^k ca^k ba^k$$

Clearly $w \notin L_D$ and $v \in L_D$. However, $F_k(\rtimes w \ltimes) = F_k(\rtimes v \ltimes)$; i.e., they have the same k-factors. In fact for any factor $f \in F_k(\rtimes w \ltimes)$, it is the case that $|w|_f = |v|_f$. Therefore $F_{k,t}(\rtimes w \ltimes) = F_{k,t}(\rtimes v \ltimes)$. If L_D were LTT, it would follow by definition that either both $w, v \in L_D$ or neither w, v belong to L_D, which is clearly false. Hence $L_D \notin$ LTT. □

Theorem 5 *TSL languages are not PT.*

Proof As above, it is sufficient to provide an example of a TSL language which is not PT. Consider any length k and the language L_D. Let

$$w = a^k(ba^k ba^k ca^k ca^k)^k \quad \text{and}$$
$$v = a^k(ba^k ca^k ba^k ca^k)^k$$

Clearly $w \notin L_D$ and $v \in L_D$. But observe that $P_{\leq k}(w) = P_{\leq k}(v)$. Hence, even though the two words have exactly the same k-subsequences (for any k), both words are not in L_D. It follows that L_D does not belong to PT. □

Although TSL languages are neither LTT nor PT, Theorem 6 establishes that they are star-free.

Theorem 6 *TSL languages are star-free.*

Proof Consider any language L which is Strictly k-Local on Tier T for some $T \subseteq \Sigma$ and $k \in \mathbb{N}$. By Theorem 2, there exists a finite set $\bar{S} \subseteq F_k(\rtimes T^* \ltimes)$ such that $L = \cap_{w \in \bar{S}} \overline{C_T(w)}$. Since the star-free languages are closed under finite intersection and complement, it is sufficient to show that $C_T(w)$ is star-free for all $w \in \rtimes T^* \ltimes$.

First consider any $w = \sigma_1 \cdots \sigma_n \in T^*$. Since $(\Sigma - T)^* = \overline{\Sigma^* T \Sigma^*}$ and $\Sigma^* = \overline{\emptyset}$, the set $C_T(w)$ can be written as

$$\overline{\emptyset}\ \overline{\overline{\emptyset} T \overline{\emptyset}}\ \sigma_1\ \overline{\overline{\emptyset} T \overline{\emptyset}}\ \sigma_2\ \overline{\overline{\emptyset} T \overline{\emptyset}}\ \cdots\ \sigma_n\ \overline{\emptyset}$$

This is a regular expression without the Kleene-star. In the cases where w begins (ends) with a word boundary symbol, the first (last) $\overline{\emptyset}$ in the GRE above should be replaced with $\overline{\overline{\emptyset} T \overline{\emptyset}}$. Since every $C_T(w)$ can be expressed as a GRE without the Kleene-star, every TSL language is star-free. □

Together Theorems 1-4 establish that TSL languages generalize the SL languages in a different way than the LT and LTT languages do (Figure 1).

4.3 Other Properties

There are two other properties of TSL languages worth mentioning. First, TSL languages are closed under suffix and prefix. This follows immediately because no word w of any TSL language contains any forbidden factors on the tier and so neither does any prefix or suffix of w. SL and SP languages–but not LT or PT ones–also have this property, which has interesting algebraic consequences (Fu et al., 2011).

Next, consider that the choice of $T \subseteq \Sigma$ and $k \in \mathbb{N}$ define systematic classes of languages which are TSL. Let $\mathcal{L}_{T,k}$ denote such a class. It follows immediately that $\mathcal{L}_{T,k}$ is a string extension class (Heinz, 2010b). A string extension class is one which can be defined by a function f whose domain is Σ^* and whose codomain is the set of all finite subsets of some set A. A grammar G is a particular finite subset of A and the language of the grammar is all words which f maps to a subset of G. For $\mathcal{L}_{T,k}$, the grammar can be thought of as the set of permissible factors on tier T and the function is $w \mapsto F_k(\rtimes E_T(w) \ltimes)$. In other words, every word is mapped to the set of k-factors present on tier T. (So here the codomain–the possible grammars–is the powerset of $F_k(\rtimes T^* \ltimes)$.)

String extension classes have quite a bit of structure, which faciliates learning (Heinz, 2010b; Kasprzik and Kötzing, 2010). They are closed under intersection, and have a lattice structure under the partial ordering given by the inclusion relation ($\subseteq$). Additionally, these classes are identifiable in the limit from positive data (Gold, 1967) by an incremental learner with many desirable properties.

In the case just mentioned, the tier is known in advance. Learners which identify in the limit a class of TSL languages with an unknown tier but known k exist in principle (since such a class is of finite size), but it is unknown whether any such learner is efficient in the size of the input sample.

5 Discussion

Having established the main results, this section discusses some implications for phonology in general, Optimality Theory in particular, and future research.

There are three classes of phonotactic constraints in phonology: local segmental patterns, long-distance segmental patterns, and stress patterns (Heinz, 2007). Local segmental patterns are SL (Heinz, 2010a). Long-distance segmental phonotactic patterns are those derived from processes of consonant harmony and disharmony and vowel harmony. Below we show each of these patterns belong to TSL. For exposition, assume Σ=$\{$*l,r,i,ö,u,o*$\}$.

Phonotactic patterns derived from attested long-distance consonantal assimilation patterns (Rose and Walker, 2004; Hansson, 2001) are SP; on the other hand, phonotactic patterns derived from attested long-distance consonantal *dis*similation patterns (Suzuki, 1998) are not (Heinz, 2010a). However, both belong to TSL. Assimilation is obtained by forbidding disagreeing factors on the tier. For example, forbidding lr and rl on the liquid tier $T = \{l, r\}$ yields only words which do not contain both [l] and [r]. Dissimilation is obtained by forbidding agreeing factors on the tier; e.g. forbidding ll and rr on the liquid tier yields a language of the same character as L_D.

The phonological literature distinguishes three kinds of vowel harmony patterns: those without neutral vowels, those with opaque vowels and those with transparent vowels (Baković, 2000; Nevins, 2010). Formally, vowel harmony patterns without neutral vowels are the same as assimilatory consonant harmony. For example, a case of back harmony can be described by forbidding disagreeing factors $\{$*iu, io, öu, öo, ui, uö, oi, oö*$\}$ on the vowel tier $T =$$\{$*i,ö,u,o*$\}$. If a vowel is opaque, it does not harmonize but begins its own harmony domain. For example if [i] is opaque, this can be described by forbidding factors $\{$*iu, io öu, öo, uö, oö*$\}$ on the vowel tier. Thus words like *lulolilö* are acceptable because *oi* is a permissible factor. If a vowel is transparent, it neither harmonizes nor begins its own harmony domain. For example if [i] is transparent (as in Finnish), this can be described by removing it from the tier; i.e. by forbidding factors $\{$*öu, öo, uö, oö*$\}$ on tier $T =$$\{$*ö,u,o*$\}$. Thus words like *lulolilu* are acceptable since [i] is not on the relevant tier. The reasonable hypothesis which follows from this discussion is that all humanly possible segmental phonotactic patterns are TSL (since TSL contains SL).

Additionally, the fact that $\mathcal{L}_{T,k}$ is closed under intersection has interesting consequences for Optimality Theory (OT) (Prince and Smolensky, 2004). The intersection of two languages drawn from the same string extension class is only as expensive as the intersection of finite sets (Heinz, 2010b). It is known that the generation problem in OT is NP-hard (Eisner, 1997; Idsardi, 2006) and that the NP-hardness is due to the problem of intersecting arbitrarily many arbitrary regular sets (Heinz et al., 2009). It is unknown whether intersecting arbitrarily many TSL sets is expensive, but the results here suggest that it may only be the intersections across distinct $\mathcal{L}_{T,k}$ classes that are problematic. In this way, this work suggests a way to factor OT constraints characterizable as TSL languages in a manner originally suggested by Eisner (1997).

Future work includes determining automata-theoretic characterizations of TSL languages and procedures for deciding whether a regular set belongs to TSL, and if so, for what T and k. Also, the erasing function may be used to generalize other subregular classes.

6 Conclusion

The TSL languages generalize the SL languages and have wide application within phonology. Even though virtually all segmental phonotactic constraints present in the phonologies of the world's languages, both local and non-local, fall into this class, it is striking how highly restricted (sub-star-free) and well-structured the TSL languages are.

Acknowledgements

We thank the anonymous reviewers for carefully checking the proofs and for their constructive criticism. We also thank the participants in the Fall 2010 Formal Models in Phonology seminar at the University of Delaware for valuable discussion, especially Jie Fu. This research is supported by grant #1035577 from the National Science Foundation.

References

Diana Archangeli and Douglas Pulleyblank. 1994. *Grounded Phonology*. Cambridge, MA: MIT Press.

Eric Baković. 2000. *Harmony, Dominance and Control*. Ph.D. thesis, Rutgers University.

Steven Bird. 1995. *Computational Phonology: A Constraint-Based Approach*. Cambridge University Press, Cambridge.

G. N. Clements and Elizabeth Hume. 1995. The internal organization of speech sounds. In John A. Goldsmith, editor, *The Handbook of Phonological Theory*, pages 245–306. Blackwell, Cambridge, Mass., and Oxford, UK.

G. N. Clements. 1976. Neutral vowels in hungarian vowel harmony: An autosegmental interpretation. In David Nash Judy Kegl and Annie Zaenen, editors, *North Eastern Linguistic Society (NELS) 7*, pages 49–64, Amherst, MA. University of Massachusetts, Graduate Linguistic Student Association.

Jason Eisner. 1997. Efficient generation in primitive Optimality Theory. In *Proceedings of the 35th Annual ACL and 8th EACL*, pages 313–320, Madrid, July.

S. Frisch, J. Pierrehumbert, and M. Broe. 2004. Similarity avoidance and the OCP. *Natural Language and Linguistic Theory*, 22:179–228.

Jie Fu, Jeffrey Heinz, and Herbert Tanner. 2011. An algebraic characterization of strictly piecewise languages. In *The 8th Annual Conference on Theory and Applications of Models of Computation*, volume 6648 of *Lecture Notes in Computer Science*. Springer-Verlag.

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

John Goldsmith and Jason Riggle. to appear. Information theoretic approaches to phonological structure: the case of Finnish vowel harmony. *Natural Language and Linguistic Theory*.

John Goldsmith. 1976. *Autosegmental Phonology*. Ph.D. thesis, MIT, Cambridge, Mass. Published by Garland Press, New York , 1979.

Gunnar Hansson. 2001. *Theoretical and typological issues in consonant harmony*. Ph.D. thesis, University of California, Berkeley.

Bruce Hayes and Colin Wilson. 2008. A maximum entropy model of phonotactics and phonotactic learning. *Linguistic Inquiry*, 39:379–440.

Jeffrey Heinz, Gregory Kobele, and Jason Riggle. 2009. Evaluating the complexity of Optimality Theory. *Linguistic Inquiry*, 40(2):277–288.

Jeffrey Heinz. 2007. *The Inductive Learning of Phonotactic Patterns*. Ph.D. thesis, University of California, Los Angeles.

Jeffrey Heinz. 2010a. Learning long-distance phonotactics. *Linguistic Inquiry*, 41(4):623–661.

Jeffrey Heinz. 2010b. String extension learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 897–906, Uppsala, Sweden, July. Association for Computational Linguistics.

William Idsardi. 2006. A simple proof that Optimality Theory is computationally intractable. *Linguistic Inquiry*, 37(2):271–275.

Anna Kasprzik and Timo Kötzing. 2010. String extension learning using lattices. In Henning Fernau Adrian-Horia Dediu and Carlos Martín-Vide, editors, *Proceedings of the 4th International Conference on Language and Automata Theory and Applications (LATA 2010)*, volume 6031 of *Lecture Notes in Computer Science*, pages 380–391, Trier, Germany. Springer.

Andras Kornai. 1994. *Formal Phonology*. Garland, New York.

John J. McCarthy. 1979. *Formal problems in Semitic phonology and morphology*. Ph.D. thesis, MIT. Published by Garland Press, New York, 1985.

Robert McNaughton and Seymour Papert. 1971. *Counter-Free Automata*. MIT Press.

Armin Mester. 1988. *Studies in Tier Structure*. New York: Garland Publishing, Inc.

Andrew Nevins. 2010. *Locality in Vowel Harmony*. The MIT Press, Cambridge, MA.

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

William Poser. 1982. Phonological representation and action-at-a-distance. In H. van der Hulst and N.R. Smith, editors, *The Structure of Phonological Representations*, pages 121–158. Dordrecht: Foris.

Alan Prince and Paul Smolensky. 2004. *Optimality Theory: Constraint Interaction in Generative Grammar*. Blackwell Publishing.

Alan Prince. 1984. Phonology with tiers. In Mark Aronoff and Richard T. Oehrle, editors, *Language Sound Structure*, pages 234–244. MIT Press, Cambridge, Mass.

James Rogers and Geoffrey Pullum. to appear. Aural pattern recognition experiments and the subregular hierarchy. *Journal of Logic, Language and Information*.

James Rogers, Jeffrey Heinz, Gil Bailey, Matt Edlefsen, Molly Visscher, David Wellcome, and Sean Wibel. 2010. On languages piecewise testable in the strict sense. In Christian Ebert, Gerhard Jäger, and Jens Michaelis, editors, *The Mathematics of Language*, volume 6149 of *Lecture Notes in Artifical Intelligence*, pages 255–265. Springer.

Sharon Rose and Rachel Walker. 2004. A typology of consonant agreement as correspondence. *Language*, 80(3):475–531.

Imre Simon. 1975. Piecewise testable events. In *Automata Theory and Formal Languages*, pages 214–222.

Keiichiro Suzuki. 1998. *A Typological Investigation of Dissimilation*. Ph.D. thesis, University of Arizona, Tucson, AZ.

Jean-Roger Vergnaud. 1977. Formal properties of phonological rules. In R. Butts and J. Hintikka, editors, *Basic Problems and Methodology and Linguistics*. Amsterdam: Reidel.