Learning tone sandhis in Structural Optimality

Izabel Ilie[1] **Andrew Lamont**[1] **Brandon Prickett**[2]
[1]University College London [2]University of Massachusetts Amherst
alexandra.ilie.23@alumni.ucl.ac.uk andrew.lamont@ucl.ac.uk
bprickett@umass.edu

Abstract

This paper examines the learnability of different types of tone sandhi in Structural Optimality (Mortensen, 2006), a constraint-based framework that posits hierarchical scales and defines constraints over the scales. Approached as a hidden structure problem, we show that Expectation Driven Parameter Learning (Jarosz, 2015) can acquire these grammars, but that their properties can make learning difficult.

1 Introduction

Learning an Optimality Theoretic grammar (Prince and Smolensky, 1993/2004) entails determining the ranking of the constraints. When the underlying and surface representations are provided, learning can be achieved using established algorithms (Tesar, 1995; Magri, 2009; Brasoveanu and Prince, 2011). However, it is often the case that surface forms are consistent with multiple, unobservable, structural representations known collectively as hidden structure. Hidden structure complicates the learning problem by requiring the learner to use their grammar to posit an unambiguous surface representation and use it to learn constraint rankings (Tesar, 1998; Tesar and Smolensky, 1998a,b, 2000). Difficulties may compound when different structural parses imply different rankings. For example, the English (eng, Indo-European) word *banana* has medial stress and is consistent with a left-aligned iamb [(.pə.ˈnæ.).ɾə.] or a right-aligned trochee [.pə.(.ˈnæ.ɾə.)]. Adopting Kager's (1999, ch. 4) metrical constraints, the former is consistent with the rankings RhType=I ≫ RhType=T and All-Ft-L ≫ All-Ft-R, as the tableau in (1) illustrates, while the latter is consistent with the opposite rankings RhType=T ≫ RhType=I and All-Ft-R ≫ All-Ft-L.

(1) *banana* with iambic footing

/pənænə/	RhType=I	RhType=T	All-Ft-L	All-Ft-R
☞ [(.pə.ˈnæ.).ɾə.]		1		1
[(.ˈpə.næ.).ɾə.]	W 1	L		1
[.pə.(.næ.ˈɾə.)]		1	W 1	L

This paper examines a different type of hidden structure which is associated globally with the grammar rather than with individual strings. Specifically, we investigate the learnability of Structural Optimality (Mortensen, 2006) which entails learning a language-specific constraint ranking as well as a language-specific scale that organizes elements hierarchically. As in the metrical example, any two mappings may imply contradictory constraint rankings or scales. Our experiments reveal that these grammars are learnable, but that successful acquisition of the correct grammar is not guaranteed. Structural Optimality is illustrated in Section 2, our learning approach is presented in Section 3, and the experiments are reported in Section 4. We discuss the results and plans for future work in Section 6.

2 Structural Optimality

Structural Optimality (Mortensen, 2006) is a theory couched in Optimality Theory (OT; Prince and Smolensky, 1993/2004) that organizes sets of segments into language-specific, hierarchical scales and defines phonotactic and (anti-)faithfulness constraints over the scales. One empirical benefit of Structural Optimality is its ability to model chain shifts including circular chain shifts which are otherwise difficult to model in OT (Moreton, 1999, 2004), but not impossible (Baković, 2007; Łubowicz, 2011; Reiss, 2023). Another is its ability to represent phonetically or phonologically arbitrary scales (Mortensen, 2006, 2.2.1).

To illustrate Structural Optimality, consider tone sandhi in Seenku (sos, Niger-Congo); we follow

Proceedings of the Society for Computation in Linguistics (SCiL) 2026, pages 329–337

McPherson in transcribing nasalization with a tilde below vowels to avoid clashing with toneal diacritics. The language contrasts four phonemic tones: superhigh (S) as in [sí̋] '*Terminalia* sp.', high (H) as in [sí] 'RECIPROCAL', low (L) as in [sì] 'first son (proper name)', and extra low (X) as in [sȉ] 'water jar' (McPherson, 2020, 80). After high-toned pronominal arguments, tone sandhi is circular X → S → H → X, as in the examples in (2). After extra low toned arguments, tone sandhi exhibits a four step chain shift S → H → X → L (3). After super high toned arguments, tones neutralize to super high {X, H, S} → S (4). There are no underlyingly low toned morphemes in the set of sandhi triggers or undergoers (McPherson, 2019, 7).

(2) Circular tone sandhi after /H/ (McPherson, 2019, 8)

X → S	/á kn̰/	[á kn̋]	'your head'
S → H	/á nɪ̋/	[á nɪ́]	'your father'
H → X	/á ɲá/	[á ɲà]	'your mother'

(3) Chain shift tone sandhi after /X/ (McPherson, 2019, 8)

S → H	/à nɪ̋/	[à nɪ́]	'his/her father'
H → X	/à ɲá/	[à ɲà]	'his/her mother'
X → L	/à kn̰/	[à kn̰]	'his/her head'

(4) Neutralizing tone sandhi after /S/ (McPherson, 2019, 8)

X → S	/mɪ̋ kn̰/	[mɪ̋ kn̋]	'our head'
H → S	/mɪ̋ ɲá/	[mɪ̋ ɲā]	'our mother'
S → S	/mɪ̋ nɪ̋/	[mɪ̋ nɪ́]	'our father'

In the Structural Optimality analysis,[1] these four tones are organized into the scale {S} < {H} < {X} < {L}, with S at the bottom, H above it, X above H, and L at the top. Each position is represented as a set to allow multiple objects to share a given position in the scale. As mentioned at the beginning of this section, scales can be phonetically or phonologically arbitrary; the scale for Seenku reflects this, as it does not correlate with the relative pitch of the tones S > H > L > X.

The analysis requires five constraints, three Structural Optimality constraints and two general phonotactic constraints. We set aside the question of how to restrict these alternations to the sandhi environments and take for granted some responsible mechanism in the grammar. The three relevant Structural Optimality constraints are defined in (5-7); we define the remaining constraints later in this section. As a simplification, we only consider grammars with single scales; Structural Optimality allows languages to employ multiple scales and defines these constraints relative to specific scales. ENDMOST is a phonotactic constraint that gradiently penalizes tones not at the bottom of the scale; i.e., [H] incurs one violation, [X] two, and [L] three. SAME is a faithfulness constraint that penalizes mappings that change steps on the scale. In this case, because each step on the scale is a singleton set, SAME penalizes all unfaithful mappings. HIGHER is an anti-faithfulness constraint that penalizes tones that do not climb the scale. SAME and HIGHER are categorical, assigning at most one violation per tone.

(5) ENDMOST (Mortensen, 2006, 32)

For every tone T, assign one violation for every step on the scale separating T from the bottom of the scale.

(6) SAME (Mortensen, 2006, 48)

For every pair of corresponding input/output tones T_i, T_o, assign one violation if T_i and T_o are not at the same step in the scale.

(7) HIGHER (Mortensen, 2006, 54)

For every pair of corresponding input/output tones T_i, T_o, assign one violation if T_o is not higher on the scale than T_i.

To these constraints, we add *S{H,L,X} which penalizes any non-super high tone immediately following a super high tone and *HL which penalizes a low tone immediately following a high tone. We do not claim that these constraints are useful elsewhere in the phonology of Seenku or cross-linguistically, but include them for the sake of presenting a complete analysis.

As the tableaux in (8-10) illustrate, the ranking {*HL, *S{H,L,X}} ≫ HIGHER ≫ ENDMOST ≫ SAME models the sandhi processes in Seenku. In the chain shift sandhi (8), the constraint HIGHER prevents underlying tones from surfacing faithfully and from descending the scale. Their ascent is kept stepwise by ENDMOST, which only tolerates minimal distance from the bottom of the scale. Ranked below both constraints, SAME disprefers unfaithful mappings, but cannot prevent their surfacing. Circular tone sandhi (9) is driven by the same pressure for tones to ascend the scale, with *HL preventing low tones from surfacing, making extra low tones

[1] This analysis was previous presented as Ilie et al. (2024).

the effective ceiling. For /HX/, no output can optimally improve on HIGHER, and the tone resets to the bottom of the scale, satisfying ENDMOST. Neutralizing sandhi (10) is driven exclusively by the constraint *S{H,L,X} which penalizes any tone other than a super high tone immediately after a super high tone.

(8) Chain shift tone sandhi after /X/

scale = {S} < {H} < {X} < {L}

		HIGHER	ENDMOST	SAME
/XS/	XS	W 1	L	L
	☞ XH		1	1
	XX		W 2	1
	XL		W 3	1
/XH/	XS	W 1	L	1
	XH	W 1	L 1	L
	☞ XX		2	1
	XL		W 3	1
/XX/	XS	W 1	L	1
	XH	W 1	L 1	1
	XX	W 1	L 2	L
	☞ XL		3	1

(9) Circular tone sandhi after /H/

scale = {S} < {H} < {X} < {L}

		*HL	HIGHER	ENDMOST	SAME
/HX/	☞ HS		1		1
	HH		1	W 1	1
	HX		1	W 2	L
	HL	W 1	L	W 3	1
/HS/	HS		W 1	L	L
	☞ HH			1	1
	HX			W 2	1
	HL	W 1		W 3	1
/HH/	HS		W 1	L	1
	HH		W 1	L 1	L
	☞ HX			2	1
	HL	W 1		W 3	1

In addition to the patterns exemplified by Seenku tone sandhi, Structural Optimality also provides a model of mappings Mortensen (2006, 2.7.3) dubs neutralization with bounce-back. An example from Hmong Sha (hmn, Hmong-Mien) is illustrated

(10) Neutralizing tone sandhi after /S/

scale = {S} < {H} < {X} < {L}

		*S{H,L,X}	HIGHER	ENDMOST	SAME
/SX/	☞ SS		1		1
	SH	W 1	1	W 1	1
	SX	W 1	1	W 2	L
	SL	W 1	L	W 3	1
/SH/	☞ SS		1		1
	SH	W 1	1	W 1	L
	SX	W 1	L	W 2	1
	SL	W 1	L	W 3	1
/SS/	☞ SS		1		
	SH	W 1	L	W 1	W 1
	SX	W 1	L	W 2	W 1
	SL	W 1	L	W 3	W 1

in (11); see Mortensen (2006, 4.2-4.3) for other examples. In these examples, tildes below vowels indicate glottalization; see Johnson and Strecker (2020, 15-17) for a detailed description of the tones, their variable realizations, and their associated laryngeal features. Three of the mappings neutralize tones 2, 4, and 8 to tone 6, reminiscent of neutralizing sandhi in Seenku (4). Rather than surfacing faithfully however, tone 6 is mapped onto tone 8, metaphorically bouncing back.

(11) Neutralizing tone sandhi with bounce-back in Hmong Sha (Johnson and Strecker, 2020, 19)

2 → 6	/sṵ˩ mpje˨˩/ ‘ear’	[sṵ˩ mpje̤˦]
4 → 6	/nti˥ tʰəʉ˨˩ te̤˦/ ‘thumb’	[nti˥ tʰəʉ˨˩ te̤˦]
8 → 6	/qa˦ naɪ̰˩/ ‘female (hen)’	[qa˦ naɪ̤˦]
6 → 8	/qa˦ nt͡ʃar˦ ma̤˦/ ‘eyeball’	[qa˦ nt͡ʃar˦ ma̰˩]

Neutralization with bounce-back is driven by the anti-faithfulness constraint DIFF (12) which penalizes mappings that are faithful with respect to the scale.

(12) DIFF (Mortensen, 2006, 49)

For every pair of corresponding input/output tones T_i, T_o, assign one violation if T_i and T_o are at the same step in the scale.

The tableaux in (13) illustrate a Structural Optimality analysis of Hmong Sha sandhi with the scale $\{6\} < \{8\} < \{2, 4\}$. In the sandhi environment (see Johnson and Strecker, 2020, 18), after a preceding tone, represented as T, tones 2, 4, and 8 surface as tone 6. The anti-faithfulness constraint DIFF rules out intra-step mappings, and ENDMOST prefers candidates with tone 6, which occupies the bottom of the scale. Because of DIFF, underlying tone 6 cannot surface faithfully. It maps onto tone 8 to minimze the violations of ENDMOST and remain as close as possible to the bottom of the scale.

(13) Neutralizing tone sandhi with bounce-back

scale = $\{6\} < \{8\} < \{2, 4\}$

		DIFF	ENDMOST	HIGHER	SAME
/T2/	☞ T6			1	1
/T4/	T8		W 1	1	1
	T2	W 1	W 2	1	L
	T4	W 1	W 2	1	L
/T8/	☞ T6			1	1
	T8	W 1	W 1	1	L
	T2		W 2	L	1
	T4		W 2	L	1
/T6/	T6	W 1	L	W 1	L
	☞ T8		1		1
	T2		W 2		1
	T4		W 2		1

For the sake of completeness, we define three additional Structural Optimality constraints in (14-16). With the four constraints defined previously, these seven constraints cover input-output mappings; see Mortensen (2006, 2.5) for related constraints defined over strings.

(14) NOHIGHER (Mortensen, 2006, 55)

For every pair of corresponding input/output tones T_i, T_o, assign one violation if T_o is higher on the scale than T_i.

(15) TOP (Mortensen, 2006, 34)

For every tone T, assign one violation if T is not at the top of the scale.

(16) EXTREME (Mortensen, 2006, 34-35)

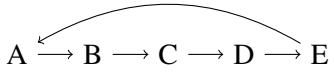
For every tone T, assign one violation if there is a step below T and a step above T.

To summarize, Structural Optimality predicts a modest typology including the five language types abstractly schematized in (17-21) following Mortensen (2006, 2.7). Additional languages where the top and bottom of the scale map onto themselves result from ranking ENDMOST relatively high; we omit these languages from discussion for brevity's sake.

(17) Chain shift

A ⟶ B ⟶ C ⟶ D ⟶ E ⮌

(18) Circular chain shift



(19) Identity

A ⮌ B ⮌ C ⮌ D ⮌ E ⮌

(20) Neutralization

(21) Neutralization with bounce-back

3 Expectation Driven Parameter Learning

Because the scales posited by Structural Optimality cannot be observed directly, they instantiate a hidden structure problem in learning. For example, if a given mapping /α/ → [β] is interpreted as the end of a chain shift, then it provides evidence for ranking HIGHER ≫ ENDMOST and evidence that α is lower on the scale than β as in (8). If the mapping is instead interpreted as neutralization, then it provides evidence for ranking ENDMOST ≫ HIGHER and evidence that β is lower on the scale than α as in (13).

Our experiments employ Expectation Driven Parameter Learning (EDPL; Jarosz, 2015), which has been shown successfully to learn grammars with hidden structure in phonology (Nazarov, 2016; Nazarov and Jarosz, 2017; Prickett and Jarosz, 2021; Nazarov and Jarosz, 2021) and syntax (Prickett et al., 2020; Hucklebridge, 2023), consistently outperforming related models. In a nutshell, EDPL

estimates the probability of some part of the grammar behaving in a particular way by randomly sampling all possible grammars that share that grammatical characteristic; the more accurate the sampled grammars, the more likely it is that that particular aspect has that behavior.

In EDPL, grammars are represented as sets of binary parameters. For example, the relative ranking of every pair of OT constraints is associated with two parameters: ψ(HIGHER, ENDMOST) represents the probability that HIGHER dominates ENDMOST, and ψ(ENDMOST, HIGHER) represents the probability that the opposite ranking holds. It is worth noting that contradictory parameters $\psi(\alpha, \beta)$ and $\psi(\beta, \alpha)$ are not defined as summing to 1, but this is an automatic consequence of learning.

Exactly like OT constraint rankings, scales in Structural Optimality are defined as directed acyclic graphs (Mortensen, 2006, 2.2). Accordingly, parallel binary parameters represent the grammar's scale. For example, ψ(2, 6) represents the probability that tone 2 occupies a lower step on the scale than tone 6, and ψ(6, 2) represents the reverse. Because learning a Structural Optimality grammar entails learning two directed acyclic graphs in parallel, it is the same computational problem as learning a stratal OT grammar (Kiparsky, 2000) with two levels. We thus based our implementation on Prickett and Jarosz (2021), who use EDPL in that context.

As a running example, consider a language with three tones, L, M, H, that participate in a chain shift L → M → H↻. As a simplification, there are only two constraints HIGHER and ENDMOST. For each pair of tones, there are two parameters representing their relative position on the scale, and there are two parameters representing the rankings of the two constraints. At the beginning of learning, these parameters can be randomized or set uniformly to 0.5 as illustrated below.

(22) Initial state of learning

ψ(L, M) = 0.5 ψ(M, L) = 0.5
ψ(L, H) = 0.5 ψ(H, L) = 0.5
ψ(M, H) = 0.5 ψ(H, M) = 0.5
ψ(ENDMOST, HIGHER) = 0.5
ψ(HIGHER, ENDMOST) = 0.5

Following Jarosz (2015, 15), the probability that a given parameter has a given value ψ_i given the current state of the grammar G and a training datum d is defined using Bayes' Law in (23). Computing the probability of the grammar generating the datum with this parameter setting $P(d|\psi_i, G)$ and the current probability of the parameter setting given the grammar $P(\psi_i|G)$ are straightforward. To compute the probability of the datum given the grammar $P(d|G)$, however, requires marginalizing over all possible parameter settings, which is intractable: there are 2^n possible grammars given binary parameters over n elements. Instead, this value is estimated by randomly sampling possible grammars according to the model's current estimate of those parameter probabilities. Sampled grammars' constraint rankings and scales were converted into a total order using topological sorting (Kahn, 1962).

(23) $P(\psi_i|G, d) = \frac{P(d|\psi_i,G)P(\psi_i|G)}{P(d|G)}$

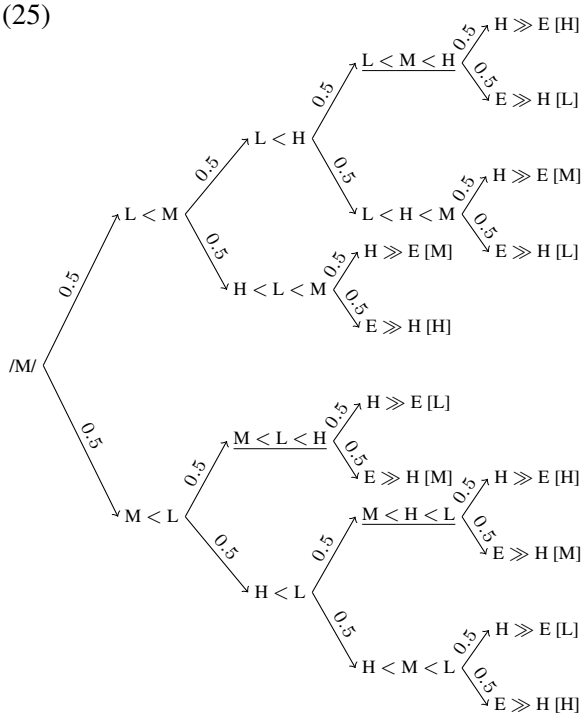
In the running example, consider the parameter ψ(M, H), that M is lower on the scale than H, and the datum /M/ → [H]. Because the number of parameters is relatively small, the entire grammar space can be represented in (25) on the next page. In the upper branch, L is lower on the scale than M, and these grammars occupy half of the probability space because ψ(L, M) = 0.5. These grammars further branch into those in which L is also lower on the scale than H or the opposite holds. In the latter, the scale parameters are all set: L is below M, H is below L, and by transitivity, H must be below L as well. At the end of each branch, the relative ranking of ENDMOST and HIGHER is represented. The surface form of /M/ is labelled at the end of each branch. Overall, the grammar correctly maps /M/ onto [H] with probability 0.3125, incorrectly produces [L] with probability 0.3125, and [M] with probability 0.375. The slight boost in faithful mappings correspond to M occupying the top or bottom of the scale. Marginalizing over grammars where M is lower on the scale than H (underlined), the probability of /M/ → [H] is 0.125. Therefore the probability of the parameter setting ψ(M, H) given this state of the grammar and the mapping /M/ → [H] is $\frac{0.125 \times 0.5}{0.3125} = 0.2$.

As the grammar space explodes in the number of parameters, the probabilities of data given the grammar (and a parameter setting) cannot be efficiently calculated and are instead estimated by randomly sampling from the possible grammars. For Structural Optimality, the space of grammars grows both in the number of constraints and the size of the scales; as far as we know, this is a unique aspect of the framework. When sampling, to calculate the probability of some datum given the grammar, we

have to marginalize over the opposite parameter setting. Thus the denominator in (23) expands to (24) following Jarosz (2015, 15). By $\neg\psi_i$, we mean the opposite parameter setting to ψ_i, for example $\neg\psi$(M, H) is ψ(H, M).

(24) $P(d|G) = P(d|\psi_i, G)P(\psi_i|G) + P(d|\neg\psi_i, G)P(\neg\psi_i|G)$

(25)



In our experiments, we estimate parameter settings using online learning, meaning that incremental updates are calculated once per data point. Online learning is parameterized by a sampling rate r and a learning rate p; in our experiments, we set r to 100 and p to 0.75. Following Jarosz (2015, 30), we operationalize online learning as follows:

(26) For a UR→SR mapping and a parameter setting ψ(A, B),

- Set A ≫ B (if A and B are constraints) or A < B (if A and B are elements on the scale) and sample 100 grammars. m_{AB} is the number of times the UR is correctly mapped onto its SR.
- Set B ≫ A or B < A and sample 100 grammars. m_{BA} is the number of times the UR is correctly mapped onto its SR.
- The expected number of correct mappings $\hat{E}_{AB}$ and $\hat{E}_{BA}$ are the number of correct mappings times the probability of that parameter setting
  $\hat{E}_{AB} = m_{AB} \times \psi$(A, B)
  $\hat{E}_{BA} = m_{BA} \times \psi$(B, A)
- The new value of the parameter setting ψ(A, B) is the weighted sum of its current value and expected number of correct mappings normalized by the two settings.
  ψ_{new}(A, B) = $0.75 \times \frac{\hat{E}_{AB}}{\hat{E}_{AB}+\hat{E}_{BA}} + 0.25 \times \psi_{current}$(A, B)
  Likewise for ψ(B, A).
  ψ_{new}(B, A) = $0.75 \times \frac{\hat{E}_{BA}}{\hat{E}_{AB}+\hat{E}_{BA}} + 0.25 \times \psi_{current}$(B, A)

Consider again the mapping /M/ → [H] and the parameter ψ(M, H) in the initial grammar. With 100 samples, we expect to sample 25 correct mappings with M < H and about 38 with H < M. With the learning rate $p = 0.75$, ψ(M, H) is lowered from 0.5 to 0.423, approaching its calculated value of 0.2, and ψ(H, M) is raised from 0.5 to 0.577. Because the actual grammar needed to model the chain shift has the scale L < M < H and the ranking HIGHER ≫ ENDMOST, this mapping incorrectly leads the learner towards a neutralization grammar with H at the bottom of the scale and the opposite constraint ranking. This makes sense because neutralization grammars are consistent with more scales than chain shift grammars and therefore occupy more of the grammar space. The mapping /L/ → [M] is inconsistent with a neutralization grammar and will push the parameter settings away from it over the course of learning. Success of the learner therefore depends on these data points working together to pull the grammar in the correct direction when their effects are averaged over the course of acquisition.

4 Experiments

Our experiments test EDPL on the five language types in (17-21) with two, three, four, and five segments each. Learning was online over 20 epochs of 100 sampled data points each. Initial pilots did not indicate much improvement with additional epochs or a higher sampling rate. All parameters were initialized to 0.5 and the learning rate was set to 0.75.[2]

[2]Our software and training data are available at https://github.com/aphonologist/EDPL_StructuralOT.

Experiments consisted of a set of input/output mappings that abstracted away from phonological substance: segments are represented by the letters A, B, etc. For example, a three segment language with a circular chain shift is represented as the set of input/output pairs {(A,B), (B,C), (C,A)}. For each input, every segment was a possible candidate, and the constraint set consisted of the Structural Optimality constraints defined in Section 2. Each run of the learner was assigned an accuracy score by averaging across the probabilities it assigned to all correct mappings. In the circular chain shift example, if /A/ mapped onto [B] with probability 0.5, /B/ onto [C] with probability 0.25, and /C/ onto [A] with probability 0.0, the learner's accuracy would be 0.25. Because EDPL is non-deterministic, we report average accuracies over 20 separate runs in each condition.

The experimental results are summarized in (27). Overall, the learner was successful, although it struggled with chain shifts defined over four segments and neutralization with bounce-back over three segments. Languages defined over two segments are ambiguous and we arbitrarily group them into categories. For example, {(A,B), (B,B)} is both a chain shift and neutralization.

(27) Summary of results by language and number of elements

	Chain shift	Circular chain shift	Identity	Neutralization	Neutralization with bounce-back
2	94.5%	87.0%	76.5%	N/A	N/A
3	63.0%	76.9%	80.1%	89.3%	47.7%
4	47.5%	77.0%	87.0%	89.8%	70.9%
5	62.9%	78.3%	92.3%	91.5%	66.8%

5 Understanding the Model's Performance

There is a trend in (27) where circular chain shifts, identity mappings, and neutralization mappings are more accurate with larger numbers of segments. For all three languages, there are multiple scales that are consistent with them. For example, the circular chain shift A $\rightarrow$ B $\rightarrow$ C $\rightarrow$ A can be defined with three scales A $<$ B $<$ C, C $<$ A $<$ B, and B $<$ C $<$ A, and in general, a circular chain shift over n segments is consistent with n scales. Neutralization mappings over n segments are consistent with $(n-1)!$ scales: the output is fixed at the bottom of the scale and the remaining $n-1$ segments can be in any order. Similarly, identity mappings do not use the scales at all, and are consistent with all $n!$ scales. By contrast, chain shifts can only be modeled with one scale.

It appears that EDPL is more successful modeling languages that are consistent with more grammars – i.e., more constraint rankings and scales for more evidence of this bias in EDPL, see Jarosz, 2016, Nazarov and Jarosz, 2021, and Prickett, 2021, 57-59. To measure this, we ran a factorial typology and calculated the R-volume of each language. R-volume is simply the total number of rankings that model a given language (Bane and Riggle, 2008; Riggle, 2010); for Structural Optimality, we also include the number of scales. The table in (28) reports the R-volumes for each of the languages in our experiments relative to the total number of grammars for that number of segments. There is a strong positive correlation (Pearson's $r = 0.677$) between learning success and R-volume suggesting that our interpretation is on the right track.

(28) Relative R-volume

	Chain shift	Circular chain shift	Identity	Neutralization	Neutralization with bounce-back
2	25.0%	25.0%	25.0%	N/A	N/A
3	0.2%	2.4%	17.9%	16.3%	3.4%
4	0.0%	0.8%	17.9%	12.3%	1.7%
5	0.0%	0.2%	17.9%	9.8%	1.0%

When the learner failed, it appeared to get stuck primarily in languages with larger R-volumes. For chain shift inputs, 24% of simulations resulted in neutralization languages and 7% resulted in circular chain shifts. 18% of circular chain shift inputs resulted in neutralization mappings. The only exceptions to this trend are the 20% of identity language inputs that resulted in neutralization mappings and the 2% that resulted in circular chain shifts.

Another possible contributing factor to the difficulty of each language is the availability and ambiguity inherent in mappings (for more on this in the context of EDPL, see Prickett et al., 2020 and Hucklebridge, 2023). Given a set of elements, every mapping /α/ $\rightarrow$ [β] is attested in the predicted

typology. For example, with β at the bottom of the scale, there is a ranking that neutralizes all segments to it. Further, the number of languages grows with the number of elements, which is unique to Structural Optimality. There are 12 predicted languages with two elements, 72 with three elements, 177 with four elements, and 570 with five elements.

Within languages, mappings are maximally ambiguous. The ambiguity of the footing of the word *banana* in Section 1 disappears once a learner encounters longer words. Intuitively, fewer grammars produce penultimate stress in *jalapeño* than in *banana*. For the Structural Optimality typology, mappings do not exhibit any such asymmetries; they are evenly dispersed in different languages both in their relative frequencies and their ranking volumes (Bane and Riggle, 2008; Riggle, 2010). Only faithful mappings are slightly under-represented: with four elements, each faithful mapping has an R-volume of 14 and an overall frequency of 0.237 compared to unfaithful mappings which have R-volumes of 15 and frequencies of 0.254 and with five elements, each faithful mapping has an R-volume of 34 (vs. 39) and a frequency of 0.179 (vs. 0.205).

Another factor contributing to difficulty of learning is the capacity for scales to be arbitrary: there is no inherent preference for any given scale. We hypothesize that including a prior for scales to be phonetically or phonologically coherent (for example, by initializing the probability of certain scale parameter settings to values other than 0.5) may improve learning by pushing the model away from certain, phonetically unmotivated grammars for an example of this kind of bias using a different approach to learning, see White, 2013 and Hayes and White, 2015. We leave for future work the question of whether imposing such a naturalness bias aids learning.

6 Discussion

In general, we found that Structural Optimality grammars were learned more often than not by our model, but that acquisition of these languages was not always guaranteed. Other learning algorithms, such as L-BFGS-B (Byrd et al., 1995) have also been adapted for hidden structure in phonology (Lee et al., 2026) and stratal, constraint-based grammars (Nazarov and Pater, 2017), so future work should explore other approaches to see if they result in higher accuracies.

Future work should also compare the results of our learning model to human behavior and typological trends. If languages that the model acquires with high accuracy correspond to languages that are easier for humans to learn or that are more common typologically, this would be strong evidence for this approach to learning hidden structure.

For now, these results show another useful application of EDPL (Jarosz, 2015) and represent a promising first step in modeling the acquisition of tonal scales.

Acknowledgments

We are grateful to three anonymous reviewers and Gaja Jarosz for their valuable feedback. All remaining errors are of course our own.

References

Eric Baković. 2007. A revised typology of opaque generalisations. *Phonology*, 24(2):217–259.

Max Bane and Jason Riggle. 2008. Three correlates of the typological frequency of quantity-insensitive stress systems. In *Proceedings of the Tenth Meeting of ACL Special Interest Group on Computational Morphology and Phonology*, pages 29–38. The Association for Computational Linguistics.

Adrian Brasoveanu and Alan Prince. 2011. Ranking and necessity: the Fusional Reduction Algorithm. *Natural Language & Linguistic Theory*, 29(1):3–70.

Richard H. Byrd, Peihuang Lu, Jorge Nocedal, and Ciyou Zhu. 1995. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208.

Bruce Hayes and James White. 2015. Saltation and the P-map. *Phonology*, 32(2):267–302.

Shay Hucklebridge. 2023. Learning and the typology of word order: a model of the Final-over-Final Condition. *Glossa: a journal of general linguistics*, 8(1).

Izabel Ilie, Juliette van Steensel, and Andrew Lamont. 2024. Sticking point: Structural Optimality cannot model tone bunnies. Poster presented at *LAGB 2024*.

Gaja Jarosz. 2015. Expectation driven learning of phonology. *Unpublished manuscript, University of Massachusetts Amherst. Available at* `https://people.umass.edu/jarosz/edl_submitted.pdf`.

Gaja Jarosz. 2016. Learning opaque and transparent interactions in Harmonic Serialism. In *Proceedings of the 2015 Annual Meeting on Phonology*, Washington, D.C. Linguistic Society of America.

Michael Johnson and David Strecker. 2020. An overview of Hmong Sha (West Hmongic, Guangnan) Phonology and Lexicon. Unpublished manuscript. Available at https://www.academia.edu/44730663/An_Overview_of_Hmong_Sha_West_Hmongic_Guangnan_Phonology_and_Lexicon.

René Kager. 1999. *Optimality Theory*. Cambridge Textbooks in Linguistics. Cambridge University Press, Cambridge.

A. B. Kahn. 1962. Topological sorting of large networks. *Communications of the ACM*, 5(11):558–562.

Paul Kiparsky. 2000. Opacity and cyclicity. *The Linguistic Review*, 17:351–365.

Seung Suk Lee, Joe Pater, and Brandon Prickett. 2026. Representing and learning stress: A MaxEnt framework for comparing learning across grammatical theories. Unpublished manuscript, University of Massachusetts Amherst. Available at https://websites.umass.edu/pater/papers/.

Anna Łubowicz. 2011. Chain shifts. In Marc van Oostendorp, Colin Ewen, Elizabeth Hume, and Keren Rice, editors, *The Blackwell Companion to Phonology*, pages 1717–1735. Wiley-Blackwell, Malden, MA.

Giorgio Magri. 2009. *A Theory of Individual-Level Predicates Based on Blind Mandatory Implicatures. Constraint Promotion for Optimality Theory*. Ph.D. thesis, Massachusetts Institute of Technology.

Laura McPherson. 2019. Seenku argument-head tone sandhi: Allomorph selection in a cyclic grammar. *Glossa*, 4(1).

Laura McPherson. 2020. *A Grammar of Seenku*, volume 83 of *Mouton Grammar Library*. De Gruyter Mouton, Berlin/Boston.

Elliott Moreton. 1999. Non-computable functions in Optimality Theory. University of Massachusetts Amherst manuscript.

Elliott Moreton. 2004. Non-computable functions in Optimality Theory. In John J. McCarthy, editor, *Optimality Theory in Phonology: A Reader*, pages 141–163. Blackwell Publishing, Malden, MA.

David Mortensen. 2006. *Logical and Substantive Scales in Phonology*. Ph.D. thesis, University of California, Berkeley.

Aleksei Nazarov. 2016. *Extending Hidden Structure Learning: Features, Opacity, and Exceptions*. Ph.D. thesis, University of Massachusetts Amherst.

Aleksei Nazarov and Gaja Jarosz. 2017. Learning parametric stress without domain-specific mechanisms. In *Proceedings of the 2016 Annual Meeting on Phonology*, Washington, D.C. Linguistic Society of America.

Aleksei Nazarov and Gaja Jarosz. 2021. The Credit Problem in parametric stress: A probabilistic approach. *Glossa: a journal of general linguistics*, 6(1).

Aleksei Nazarov and Joe Pater. 2017. Learning opacity in Stratal Maximum Entropy Grammar. *Phonology*, 34(2):299–324.

Brandon Prickett. 2021. *Learning Phonology with a Sequence-to-Sequence Neural Network*. Ph.D. thesis, University of Massachusetts Amherst.

Brandon Prickett, Kaden Holladay, Shay Hucklebridge, Max Nelson, Rajesh Bhatt, Gaja Jarosz, Kyle Johnson, Aleksei Nazarov, and Joe Pater. 2020. Learning syntactic parameter settings without triggers by assigning credit and blame. In *Proceedings of the Fifty-fifth Annual Meeting of the Chicago Linguistic Society*, pages 337–350, Chicago, IL. Chicago Linguistic Society.

Brandon Prickett and Gaja Jarosz. 2021. Modeling the acquisition of phonological interactions: Biases and generalization. In *Supplemental Proceedings of the 2020 Annual Meeting on Phonology*, Washington, D.C. Linguistic Society of America.

Alan Prince and Paul Smolensky. 1993/2004. *Optimality Theory: Constraint Interaction in Generative Grammar*. Blackwell Publishing, Malden, MA.

Charles Reiss. 2023. A simple account of chain shifts and saltations in Optimality Theory. Unpublished manuscript, Concordia University.

Jason Riggle. 2010. Sampling rankings. Unpublished manuscript, University of Chicago.

Bruce Tesar. 1995. *Computational Optimality Theory*. Ph.D. thesis, University of Colorado.

Bruce Tesar. 1998. An iterative strategy for language learning. *Lingua*, 104(1-2):131–145.

Bruce Tesar and Paul Smolensky. 1998a. Learnability in Optimality Theory. *Linguistic Inquiry*, 29(2):229–268.

Bruce Tesar and Paul Smolensky. 1998b. Learning Optimality-Theoretic grammars. *Lingua*, 106(1-4):161–196.

Bruce Tesar and Paul Smolensky. 2000. *Learnability in Optimality Theory*. The MIT Press, Cambridge, MA.

James White. 2013. *Bias in Phonological Learning: Evidence from Saltation*. Ph.D. thesis, University of California, Los Angeles.