Tier Alphabet(s): Tendencies and Restrictions

Alëna Aksënova

April 20, 2017

Abstract

The subregular class of Tier-based Strictly Local languages provides a good fit for most of the natural language phonology. Its core idea is to capture long-distance dependencies by projecting the set of elements involved in a phonological process on a special *tier*, in order to establish a local relation among those remote units. This paper investigates the nature of the sets of elements involved in a long-distance agreement, i.e. tendencies and restrictions applied to the tiers. I give an overview of the long-distance harmonies that involve spreading of more than one feature, and show that there are certain configurations that are avoided. Mostly, vowel harmonies require just a single tier; consonantal (sibilant) harmonies need either a single tier, or two tiers, alphabets of which are in the set-subset relation; separate vowel and consonantal harmonies are captured by the tier-based strictly local grammars, tier alphabets of which are disjoint. Interestingly, there are no attested patterns where the tier alphabets only partially overlap. This topic is relevant for both for phonological theory and computational approach: apart from revealing new typological generalizations, it brings naturalness conditions to the formal grammars applied to the language.

1 Introduction

In phonology, we often observe that certain sets of elements exhibit some particular behavior within a domain. Harmony can be used as one of the clearest examples of the process that affects segments only of a certain type: vowels, nasals, sibilants, liquids, etc. Each of these names denotes a *class* of sounds, i.e. some set that includes them. However, the sole existence of natural classes does not provide any restrictions, as was also noted by Flemming (2005) – it does not actually exclude any phonological processes. Here, my goal is to find the interaction between the sets of sounds and possible processes affecting them. As the litmus test that helps to unveil the configurations observed in natural languages, I use harmonic systems that involve spreading of more than one feature, further referred as *double harmonies*.

This topic lays in-between the two types of universals introduced by Chomsky (1965) in order to define the two major parts of knowledge needed for the description of languages. *Substantive universals* concern the instruments we need for the linguistic description, such as theoretic assumptions, types of rules or generalizations, or any other mechanisms relevant all natural languages. *Formal universals* involve the concrete character of the rules or concepts, such as the specific rules needed for capturing phenomena of a concrete language. Katz and Postal (1964) define this distinction as the one "between the form of [...] statements and their content". When talking about the sets of items involved in long-distance dependencies, their possible relations reveal the inventory of choices we have when acquiring phonological patterns. This is directly related to the notion of the substantive universals discussed above: substantive universals describe what is possible and what is not for the natural language, and the restrictions on the tier alphabets is a part of it. However, the concrete choice of items exhibiting certain harmonic pattern is a matter of formal universals, because it appears to be language-specific. Investigating the case of double harmonic patterns helps to understand better what structures are available in the natural languages, therefore telling us more about the desired universal grammar.

For decades, it has been known that the class of regular languages allows us to describe natural language phonology (Kaplan and Kay 1994). Later, Heinz (2011) showed that in fact, the whole power of regular languages is not needed, therefore the phonological analyses can be made in terms of the weaker, *subregular* classes. The descriptions here are provided in terms of the subregular class of tier-based strictly local (TSL) languages that can capture long-distance dependencies locally by projecting elements relevant for a certain process on a tier, therefore "ignoring" all the intervening irrelevant material. The TSL approach, while being sufficient enough, solves many overgeneration problems of regular languages (Gainor et al. 2012). But even though it is much less powerful, it still allows the random choice of the elements that are projected on the tier. Here I explore the restrictions on these sets – the *tier alphabets*, and these restrictions are one more step towards bringing naturalness to formal grammars.

Intuitively, it seems that a separate tier is needed for each one of the harmonies involved in a double harmonic process. In this paper, I show that this is not always true: in the case of vowel harmonies, both feature spreading processes can be fit on the same tier. However, if more than one tier-based strictly local grammar is needed, there are three logical possibilities of the tier alphabets relations: they can be either in set-subset relations, or totally disjoint, or only partially overlap. Evidence from natural languages shows that the double consonantal harmonies either require just a single tier, or fall into the first type of the multiple tier list, separate vowel and consonantal assimilation processes fall into the second one, and the partially overlapping sets appear to be non-existent.

In this paper, I give an overview of such double harmonic systems and discuss tendencies and restrictions applied to tier alphabets. Sec. 2 briefly introduces the paradigm of the TSL phonology that I follow in this paper. In Sec. 3, I define the notion of *double harmony* that is crucial for the current work, and discuss computational characteristics of such double harmonic systems. Sec. 4 provides a summary table showing harmonies in different languages that fall into each of the previously defined categories. The last section concludes the paper. Additional data and proofs are contained in the Appendix A, B, and C.

2 Subregular Manner of Handling Phonology

Recent investigations in the field of computational phonology let us reduce the class of formal languages required for the generation of natural language phonological patterns from regular languages (Kaplan and Kay 1994) to the *subregular* ones (Heinz et al. 2011, 2012). Besides the ability of subregular languages to capture the existent patterns as well as it was possible with the regular ones, they help to avoid many over-predicted though non-existent patterns, see Gainor et al. (2012) for numerous examples. As is claimed by Heinz et al. (2011), most of the phonological patterns fit in the two following subregular language classes: Strictly Local (SL) and Tier-based Strictly Local (TSL) ones.

2.1 Strictly Local Grammars

The core idea behind Strictly Local grammars is to filter strings containing substrings that must not be allowed in the language. A SL grammar consists of the alphabet Σ – the set of symbols that is used by the language, and the set of n-grams (substrings of length n) G_{SL} that must not be found in a well-formed string of the language.

For example, in RUSSIAN, obstruents in a cluster must agree in voicing, see the data below in (1-2).

(1)	[s]-tajat^j	'PREF-melt'	*[z]-tajat^j
(2)	[z]-bit^j	'PREF-beat'	*[s]-bit^j

The telic prefix *s-* is realized as [s] if followed by a voiceless consonant (1), and as [z] if followed by a voiced one (2). In this case, the alphabet $\Sigma = \{$s, z, t, b, j, a, i$\}$. The grammar prohibits combinations of obstruents with mixed voicing: $G_{SL} = \langle$*sz, *zs, *sb, *bs, *zt, *tz, *tb, *bt$\rangle$. Such grammar then would correctly rule out the forbidden forms of the words given above, where the prefix disagrees in voicing with the following obstruent.

As another phonological process, consider vowel harmony in LOKAA (Niger-Congo). In this language, a non-high vowel agrees with the preceding non-high vowel in ATR, whereas all high vowels and consonants are transparent for the harmony. See the SPE-style rule in (A1) and the examples below from (Akinlabi 2009).

(A1) $\begin{bmatrix} -\text{consonantal} \\ -\text{high} \end{bmatrix} \rightarrow \begin{bmatrix} \alpha\ \text{tense} \end{bmatrix} \ / \begin{bmatrix} -\text{consonantal} \\ -\text{high} \\ \alpha\ \text{tense} \end{bmatrix} \quad ... \quad \_\_$

(3)	èsìsòn	‘*smoke*’	*èsìsɔ̀n
(4)	ɛ̀sísɔ̀n	‘*housefly*’	*ɛ̀sísòn
(5)	lèjìmə̀	‘*matriclan*’	*lèjìmà
(6)	ɛ́kílìkà	‘*kind of plant*’	*ékílìkà

In this case, the agreeing items are not adjacent to each other. For strings such as (3) or (4), we need 5-grams to capture this pattern, because there are 3 intervening elements in-between the two agreeing non-high vowels. But for (6), even this window size is not enough: there are 5 segments in-between *ɛ* and *a*. In this language, there is no upper bound on the amount of material separating the two non-high vowels agreeing with respect to the [tense] feature, therefore no SL grammar can be constructed to capture this pattern[1]. This problem is solved by the Tier-based Strictly Local grammars.

2.2 Tier-based Strictly Local Grammars

Tier-based Strictly Local grammars (Heinz 2011, Heinz et al. 2011) capture non-local dependencies by projecting elements of a certain type on the *tier* in order to achieve locality among the remote units. Then a long-distance process can be viewed as local over the tier, because all the intervening material in-between the two dependent elements is ignored. As is done in SL grammars, illicit sequences of segments can be ruled out in a local fashion over the tier. TSL grammar consists of the tier alphabet T that is a subset of the alphabet of Σ, and the set of n-grams G_{TSL} that must not be found on a tier representation of a well-formed string.

LOKAA vowel harmony described in the previous subsection affected arbitrary far non-high vowels and only those, so it is not possible to construct a strictly local grammar capturing this process. But the power of TSL languages allows us to project remote dependent items on a tier, and block illicit combinations over this tier. In the table below, I show the feature configurations of the tier-adjacent non-high vowels under which a bigram is considered illicit.

In this case, the tier alphabet includes all non-high vowels presented in the language, and the ATR spreading is by simply blocking the combinations of non-high vowels disagreeing in their [tense] specifications, see H_{ATR}. The figure 1 illustrates this analysis.

[1] See appendix B for the discussion of the *suffix substitution closure*.

	Tier of non-high vowels T = {ɛ, e, o, ə, ɔ, a}
1.	*[α tense] [β tense]
	H_{ATR} = {*ɛe, *eɛ, *ɛo, *oɛ, *ɛə, *əɛ, *ɔe, *eɔ, *ɔə, *əɔ, *ɔo, *oɔ, *aə, *əa, *ao, *oa, *ae, *ea}

Table 1: TSL grammar for LOKAA harmony

Figure 1: ATR harmony in LOKAA

The left subfigure shows the well-formed word *èsìsòn*. The only non-high vowels (*e* and *o*) are projected on the tier, and their combination *eo* is not among those that need to be ruled out, and thus the word *èsìsòn* is considered acceptable. Another subfigure shows the ill-formed word **èsìsɔ̀n*, where the two non-high vowels *è* and ɔ̀ disagree in their [tense] specifications. After *è* and ɔ̀ are projected on a tier, we see that the bigram **eɔ* is banned by the grammar, and therefore this word must be ruled out.

3 Computational Characteristics of Double Harmonies

In this section, I use the previously discussed TSL approach to capture different harmonic patterns. A harmonic process picks out a set of elements (*undergoers*) and establishes an agreement relation among them with respect to a certain feature. This agreement might be blocked: there are some items (*blockers*) that can stop this spreading, so the previous harmonizing element will not affect the following one if a blocker is found in-between them. All other segments are *transparent* for the long-distance assimilation – they do not affect and are unaffected by the harmonic process.

I refer to any long-distance agreement process in a language that involves spreading of more than one feature as *double harmony*. For example, in many languages, vowel harmony in a feature such as backness (TURKISH, FINNISH) or tongue root position (MONGOLIAN, BURYAT) co-exists with the labial assimilation, see (Kaun 1995) for numerous examples of double vowel harmonies. Or it can be sibilant harmony in two features such as anteriority and voicing (NAVAJO, TUAREG). Also in several languages it is possible to find both consonantal and vowel harmonies (KIKONGO, KIYAKA, BUKUSU).

The results show that double vowel harmonies require just a single tier, even though more than one feature is being transmitted. For sibilant harmonies, one tier is not enough, therefore two tiers need to be projected, and their tier alphabets are in a set-subset relation. Languages with both vowel and consonantal harmonies also require two tiers, but in this case the requirement for those is not to share any common element in their tier alphabets, i.e. they need to be disjoint. So far, there are no attested cases of intersecting alphabets, when the sets of items involved in the two spreadings only partially overlap.

Sec. 3.1 shows that double vowel harmonies can be captured via just a single tier using KIRGHIZ, BURYAT, and YAKUT patterns as the examples[2]. Sec. 3.2 discusses double sibilant harmonies that sometimes 'fit' on the same tier (TUAREG), but also might have two tiers with embedded tier alphabets, that is when one spreading operates over the subset of elements taking part in the other one: IMDLAWN TASHLHIYT. In Sec. 3.3, I provide examples of the disjoint tier alphabets, where the harmonic processes operate over absolutely different sets of segments: KIKONGO, BUKUSU. The example of the unattested pattern that would require two grammars with intersecting tiers is given in Sec. 3.4.

3.1 Double Vowel Harmonies

The naïve expectation is that each of the harmonic features of the double harmony will require its own tier. However, the data shows that in the case of double vowel harmony, both harmonies fit on the same tier. Note that it does not mean that undergoers and blockers are the same for both harmonies, it only means that none of the items taking part in one harmony is irrelevant for the other one. In most of the cases, one of the harmonies affects all vowels, and the other one spreads its feature only among the tier-adjacent [α high] vowels, whereas [β high] ones function as blockers (TURKISH, MONGOLIAN, BURYAT, TATAR). Sometimes two features are transmitted simultaneously (KIRGHIZ). In other cases, all vowels function as undergoers with respect to both harmonies, while only a subset of those vowels can further spread both harmonic features – the others function as harmonizing blockers (YAKUT). See (Kaun 1995) for numerous examples of labial feature spreadings that commonly serve as one of the two harmonies in a double harmonic system. This makes it possible to formulate two main requirements for the abstract harmonies A and B in order to fit them on the same tier:

- The set of items transparent for the harmonies A and B must be identical;
- Both blockers and undergoes for the harmony A must play some role in the harmonic process B, i.e. they must be either blockers or undergoers.

[2]The only 'problematic' vowel harmony that requires adopting the mechanism of *structural projection* (De Santo and Graf 2017) is discussed in the Appendix A.

Here I show three examples of such double spreadings: fronting and rounding harmony in KIRGHIZ, where two features are transmitted together, i.e. the set of undergoers is the same for both spreadings; ATR and rounding harmony in BURYAT, where some undergoers for the ATR harmony function as blockers for the labial one; and fronting and labial harmony in YAKUT, where some harmonizing items block the spreading in certain configurations.

3.1.1 Kirghiz (Turkic)

The rule of double vowel harmony in KIRGHIZ[3] (Turkic) is to spread the features of frontness and rounding simultaneously: all vowels within a word must agree in these two features, see the rule in (B1). A locative affix with a non-high vowel (*-de, -dö, -da, -to*) and a genitive affix with a high vowel (*-nɨn, -din, -dün, -tun*) are used in the examples (5-12) below from (Nanaev 1950).

(B1) $\left[\text{– consonantal}\right] \rightarrow \begin{bmatrix}\alpha\text{ front}\\ \beta\text{ round}\end{bmatrix} / \begin{bmatrix}\text{– consonantal}\\ \alpha\text{ front}\\ \beta\text{ round}\end{bmatrix}$ —

(5)	kɨz-da	'girl-LOC'	(9)	kɨz-nɨn	'girl-GEN'
(6)	ot-to	'fire-LOC'	(10)	ot-tun	'fire-GEN'
(7)	kim-de	'who-LOC'	(11)	kim-din	'who-GEN'
(8)	üj-dö	'house-LOC'	(12)	üj-dün	'house-GEN'

There are two harmonizing features that can be either plus or minus, therefore all vowels within a word have a choice among the following four possibilities: [+front, +round], [+front, –round], [–front, +round], and [–front, –round]. For each of these options, there is an example in the data above that shows that affixes agree with the root

Vowel tier	
	T = {a, ɨ, e, i, o, ö, u, ü}
1.	*[α front] [β front]
	H_{front} = {*ai, *ae, *aö, *aü, *oi, *oe, *oö, *oü, *ɨi, *ɨe, *ɨö, *ɨü, *ui, *ue, *uö, *uü, *ia, *io, *iɨ, *iu, *ea, *eo, *eɨ, *eu, *öa, *öo, *öɨ, *öu, *üa, *üo, *üɨ, *üu}
2.	*[α round] [β round]
	H_{round} = {*oa, *oɨ, *ua, *uɨ, *öi, *öe, *üi, *üe, *ao, *au, *iö, *iü, *ɨo, *ɨu, *eö, *eü, *oe, *oi, *öa, *öɨ, *ue, *ui, *üa, *üɨ, *eo, *io, *aö, *ɨö, *eu, *iu, *aü, *ɨü}

Table 2: TSL grammar for KIRGHIZ harmony

[3]Here I present the data of the southern dialect of KIRGHIZ, see (Herbert and Poppe 1963) and (Kaun 1995) for other dialects of this language.

in these two parameters. For instance, fronted rounded vowel in *üj* 'house' spreads its feature specifications both to non-high (8) and high (12) affixes. The table below shows the feature configurations that must be avoided in the well-formed vowel sequence of KIRGHIZ words: any bigram where the two vowels disagree in fronting or rounding must be blocked.

Both spreadings operate over the same set of segments, therefore the tier alphabet T consists of all the vowels. To enforce the frontness harmony, combinations of vowels disagreeing in their [front] specification must be blocked by H_{front}. In addition to it, substrings with vowels of different [round] specification also need to be ruled out by H_{round}. Note that only one TSL grammar is needed to capture this pattern. Its tier alphabet is T, and the grammar G_{TSL} is a combination of H_{front} and H_{round} sets of the banned bigrams.

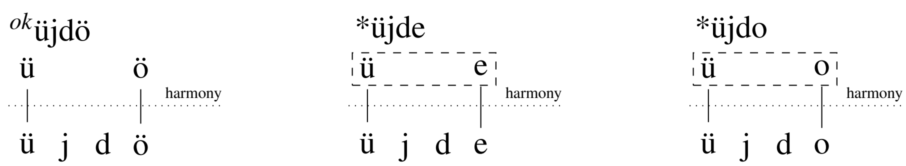


Figure 2: Frontness and labial harmony in KIRGHIZ

Consider Figure 2 as the graphic example of the grammar constructed above in action. The left subfigure represents the well-formed word *üjdö*. Both vowels are projected on a tier, and nothing is blocked, because the combination *üö* is permitted: both *ü* and *ö* agree in their fronting and rounding features. The middle subfigure shows a violation of the rounding harmony, wherefore bigram **üe* is ruled out by the H_{round} part of the grammar. Likewise, the fronting harmony violation (see the right subfigure) is ruled out by H_{front}, because **üo* is listed as an ill-formed substring. In KIRGHIZ, the set of undergoers for one harmonic spreading is exactly the same as for the other one, so, as a result, only one TSL grammar is needed to capture the desired pattern.

3.1.2 Buryat (Mongolian)

In BURYAT (Mongolian), all vowels within a word must agree in ATR. All tier-adjacent non-high vowels agree in rounding, unless there is an intervening high vowel that blocks this assimilation. As a result, the only possible positions in which rounded non-high vowels can be found are the initial syllables, and in other positions if they are licensed by the labial spreading (Poppe 1960). The set of transparent items is the same for both harmonies, and includes the only non-rounded high vowel /i/ and all consonants, see (van der Hulst and Smith 1987, Skribnik 2003, Svantesson et al. 2005) for details. The rules in (C1-3) summarize this pattern.

$$\text{(C1)} \begin{bmatrix} -\text{consonantal} \end{bmatrix} \rightarrow \begin{bmatrix} \alpha\ \text{tense} \end{bmatrix} / \begin{bmatrix} -\text{consonantal} \\ \alpha\ \text{tense} \end{bmatrix} \quad (\,/\text{i}/\,) \quad \_\_$$

$$\text{(C2)} \begin{bmatrix} -\text{consonantal} \\ -\text{high} \end{bmatrix} \rightarrow \begin{bmatrix} \beta\ \text{round} \end{bmatrix} / \begin{bmatrix} -\text{consonantal} \\ -\text{high} \\ \beta\ \text{round} \end{bmatrix} \quad (\,/\text{i}/\,) \quad \_\_$$

$$\text{(C3)} \begin{bmatrix} -\text{consonantal} \\ -\text{high} \end{bmatrix} \rightarrow \begin{bmatrix} -\text{round} \end{bmatrix} / \begin{bmatrix} -\text{consonantal} \\ +\text{high} \end{bmatrix} \quad (\,/\text{i}/\,) \quad \_\_$$

In the examples below, I illustrate the harmony process using the following two affixes: the causative suffix *-ʊ:l, -u:l*, where the vowel is specified as high, therefore agrees with the stem only in ATR; and the perfective affix *-a:d, -ɔ:d, -e:d, -o:d* with a non-high vowel that agrees with the preceding segment in ATR and, if that segment is also non-high, in rounding.

(13)	ɔr-ɔ:d	‘enter-PERF’	*ɔr-a:d
(14)	ɔr-ʊ:l-a:d	‘enter-CAUS-PERF’	*ɔr-ʊ:l-ɔ:d
(15)	to:r-o:d	‘wander-PERF’	*to:r-e:d
(16)	to:r-u:l-e:d	‘wander-CAUS-PERF’	*to:r-u:l-o:d
(17)	mɔrin-ɔ:	‘horse-POSS’	*mɔrin-a:
(18)	o:rin-go:	‘group-POSS’	*o:rin-ge:

In (13), the non-high perfective affix agrees with the non-high root vowel in ATR and rounding: both vowels are lax and rounded. But adding the high causative affix in-between them, as in (14), results in the blocking of the labial spreading: the perfective affix no longer agrees with the stem in rounding, because they are separated from each other by an intervening high vowel. Examples (15, 16) show the same effect for the tense root. Examples (17) and (18) show the transparency of the intervening vowel /i/.

Vowel tier (except /i/) T = {a, e, ɔ, o, ʊ, u}	
1.	*[α tense] [β tense]
	H_{ATR} = {*ɔo, *oɔ, *ʊu, *uʊ, *ae, *ea, *ʊo, *oʊ, *ʊe, *eʊ, *ɔu, *uɔ, *ɔe, *eɔ, *oa, *ao, *au, *ua}
2.	*[– high, α round] [– high, β round]
	H_{r1} = {*ɔa, *aɔ, *eo, *oe, *ao, *oa, *eɔ, *ɔe}
3.	*[+ high] [– high, + round]
	H_{r2} = {*ʊɔ, *uo, *uɔ, *ʊo}

Table 3: TSL grammar for BURYAT harmony

The set of elements involved in each of these two spreadings is the same, so the tier alphabet T consists of all relevant vowels for harmony. H_{ATR} blocks combinations of vowels that disagree in tense, and $H_{r1} + H_{r2}$ enforce the tier-adjacent non-high vowels to harmonize in rounding, and also block rounded non-high vowels after the high ones. Again, only one TSL grammar is needed: its tier alphabet is T, and $G_{TSL} = H_{ATR} + H_{r1} + H_{r2}$.

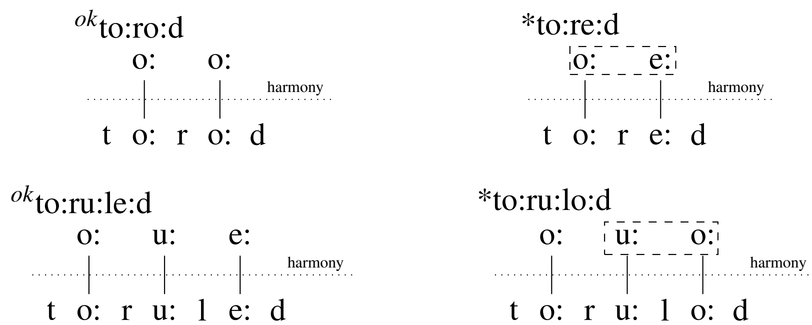


Figure 3: ATR and labial harmony in BURYAT

The first line of a figure 3 shows that the tier-adjacent non-high vowels must agree in rounding, therefore the tier bigram *oo* is allowed, whereas illicit substrings such as **oe* are ruled out by the grammar H_{r1}. But if there is an intervening high vowel in-between them, as on the second line of the figure 3, labial harmony does not happen: occurrence of the rounded non-high vowel after the high one (**uo*) is also blocked by the grammar. As this analysis shows, BURYAT double vowel harmony also requires just a single tier, because both spreadings operate over the same alphabet.

3.1.3 Yakut (Turkic)

In YAKUT (Turkic), all vowels must agree in fronting. Labial harmony spreads from the low vowels onto both low and high ones, from the high vowels to the high ones, but it cannot spread from the high vowels to the low ones. The latter ones in this case function as *harmonizing blockers*: they harmonize with any previous vowel, but block the rounding assimilation in [+high][–high] configuration. See (Sasa 2001, 2009) for details and an OT account. This pattern is outlined in (D1-4).

(D1) $\left[\ -\text{consonantal}\ \right] \rightarrow \left[\ \alpha\ \text{front}\ \right]\ /\ \begin{bmatrix} -\text{consonantal} \\ \alpha\ \text{front} \end{bmatrix}\ \ \_\_$

(D2) $\begin{bmatrix} -\text{consonantal} \\ +\ \text{high} \end{bmatrix} \rightarrow \left[\ \beta\ \text{round}\ \right]\ /\ \begin{bmatrix} -\text{consonantal} \\ \beta\ \text{round} \end{bmatrix}\ \ \_\_$

$$\text{(D3)} \begin{bmatrix} -\text{consonantal} \\ -\text{high} \end{bmatrix} \rightarrow \begin{bmatrix} \gamma\,\text{round} \end{bmatrix} / \begin{bmatrix} -\text{consonantal} \\ -\text{high} \\ \gamma\,\text{round} \end{bmatrix} \quad \_\_$$

$$\text{(D4)} \begin{bmatrix} -\text{consonantal} \\ -\text{high} \end{bmatrix} \rightarrow \begin{bmatrix} -\text{round} \end{bmatrix} / \begin{bmatrix} -\text{consonantal} \\ +\text{high} \end{bmatrix} \quad \_\_$$

The accusative affix *-(n)ü, -(n)u, -(n)ɨ, -(n)i* with a high vowel and the plural marker *-lor, -lör, -lar, -ler* with a non-high vowel demonstrate this pattern, see the examples (19-26) below from (Kaun 1995).

(19)	oɣo-lor	'child-PL'	*oɣo-lar
(20)	börö-lör	'wolf-PL'	*börö-ler
(21)	oɣo-nu	'child-ACC'	*oɣo-nɨ
(22)	börö-nü	'wolf-ACC'	*börö-ni
(23)	murum-u	'nose-ACC'	*murum-ɨ
(24)	tünnük-ü	'window-ACC'	*tünnük-i
(25)	ojum-lar	'shaman-PL'	*ojum-lor
(26)	tünnük-ler	'window-PL'	*tünnük-lör

High suffixal vowels agree with any preceding vowel in both fronting and rounding (21-24), whereas low vowels while always agree in fronting, and agree in rounding only when the preceding vowel is low (19, 20), otherwise they are unrounded (25, 26).

Vowel tier **T = {a, ɨ, e, i, o, ö, u, ü}**	
1.	*[α front] [β front]
	H_{front} = {*ai, *ae, *aö, *aü, *oi, *oe, *oö, *oü, *ɨi, *ɨe, *ɨö, *ɨü, *ui, *ue, *uö, *uü, *ia, *io, *iɨ, *iu, *ea, *eo, *eɨ, *eu, *öa, *öo, *öɨ, *öu, *üa, *üo, *üɨ, *üu}
2.	*[+ high, α round] [+ high, β round]
	H_{r1} = {*uɨ, *üi, *iü, *ɨu, *iu, *ɨü, *ui, *üɨ}
3.	*[+ high, α round] [– high, + round]
	H_{r2} = {*üö, *uo, *iö, *ɨo, *io, *ɨö, *uö, *üö}
4.	*[– high, α round] [β round]
	H_{r3} = {*oa, *oɨ, *öi, *öe, *ao, *au,*eö, *eü, *aö, *aü, *eo, *eu, *oi, *oe, *öa, *öɨ}

Table 4: TSL grammar for YAKUT harmony

The tier alphabet T consists of all YAKUT vowels. H_{front} rules out sequences of vowels that disagree in fronting, whereas $H_{r1} + H_{r2} + H_{r3}$ blocks occurrence of a rounded low vowel if it is preceded by a high one, and also any other combination of vowels that disagree in their labial features. The obtained TSL grammar operates over the tier alphabet T and its list of illicit substrings is $G_{TSL} = H_{front} + H_{r1} + H_{r2} + H_{r3}$.

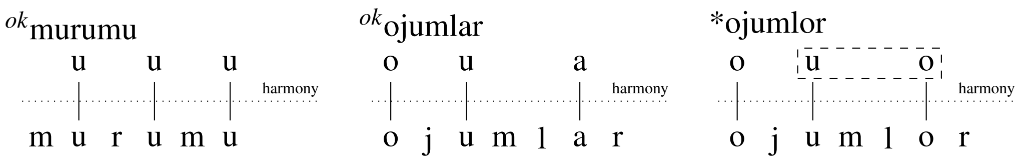


Figure 4: Fronting and labial harmony in YAKUT

Figure 4 shows that under this analysis, the word *murumu* is well-formed – the labial harmony spreads among high vowels. It also spreads from the non-high vowel *o* to the following high vowel *u* in the word *ojumlar*. However, it cannot spread from a high vowel to a low one, so **ojumlor* is blocked as the illicit bigram **uo* is found on its vowel tier. As in the previous cases, a single tier is enough to capture YAKUT double vowel harmony.

3.1.4 Double vowel harmonies: summary

In the previous three subsections, I showed three examples of double vowels harmonies, and for all of them, only one TSL grammar was needed in order to capture the pattern. It was possible because neither of the vowels taking part in one spreading was irrelevant for the other harmony. Briefly, the discussed patterns were the following.

1. **Kirghiz:** all vowels within a word harmonize in both [front] and [round] features, so all vowels are undergoers for the both spreadings.

2. **Buryat:** all vowels within a word harmonize in [tense], and all non-high vowels agree with respect to the [round] feature, whereas the intervening high vowels block this spreading. As the result, all vowels are undergoers for the [tense] harmony, additionally, non-high vowels are undergoers for the labial one, and the high vowels function as blockers in the latter case.

3. **Yakut:** all vowels harmonize in [front] and [round] except for the configuration when a high vowel precedes the non-high one: in this case, the labial spreading is blocked, whereas the fronting harmony is unaffected.

3.2 Double Sibilant Harmonies

In the previous section, I showed that one tier is enough for double vowel harmonies. Although there are double sibilant harmonies that also require just a single tier, as the TUAREG pattern that is discussed below, there are cases when it is impossible to avoid projection of two tiers. In this case, their tier alphabets are in the set-subset relation. For example, in IMDLAWN TASHLHIYT, sibilants agree in voicing in anteriority. Whereas the voicing assimilation can be blocked by an intervening voiceless obstruents, anteriority harmony is not blocked by anything, and thus the set of segments involved in the voicing harmony is the superset of the ones taking part in the anteriority spreading.

3.2.1 Tuareg (Berber)

In TUAREG (Berber), sibilants regressively agree with the following sibilant in voicing and anteriority, otherwise they are realized as [s], see (Hansson 2010b). This pattern is outlined below in (E1-2) and demonstrated in (27-31) by the causative prefix *s*-.

$$\text{(E1)} \begin{bmatrix} +\text{ consonantal} \\ +\text{ cont} \end{bmatrix} \rightarrow \begin{bmatrix} \alpha \text{ anterior} \\ \alpha \text{ voice} \end{bmatrix} \; / \; \_\_ \; \begin{bmatrix} +\text{ consonantal} \\ +\text{ cont} \\ \alpha \text{ anterior} \\ \alpha \text{ voice} \end{bmatrix}$$

$$\text{(E2)} \begin{bmatrix} +\text{ consonantal} \\ +\text{ cont} \end{bmatrix} \rightarrow \text{[s]} \; / \; \text{elsewhere}$$

(27) s-əlməd ‘CAUS-learn’
(28) s-əq:usət ‘CAUS-inherit’
(29) z-əntəz ‘CAUS-extract’
(30) ʃ-əm:əʃən ‘CAUS-be.overwhelmed’
(31) ʒ-ək:uʒət ‘CAUS-saw’

In (27), there are no sibilants in the root, therefore the causative prefix is realized in its underlying form [s]. In all other cases, this affix agrees in both voicing and anteriority with the sibilant in the stem, therefore it can be realized as [–voice, +ant] (28), [+voice, +ant] (29), [–voice, –ant] (30), and [+voice, –ant] (31).

Sibilant tier **T = {s, z, ʃ, ʒ}**	
1.	*[α ant] [β ant]
	H_{ant} = {*sʃ, *sʒ, *ʃs, *ʒs, *zʃ, *ʃz, *ʒz, *zʒ}
2.	*[α voice] [β voice]
	H_{voice} = {*sz, *zs, *ʃz, *ʃʒ, *ʒʃ, *sʒ, *ʒs, *zʃ}

Table 5: TSL grammar for TUAREG harmony

Both spreadings operate over the tier of sibilants T, and other segments are not involved in the assimilation. The grammar blocks any combination of sibilants that disagree in voicing or anteriority: $G_{sib} = H_{ant} + H_{voice}$.

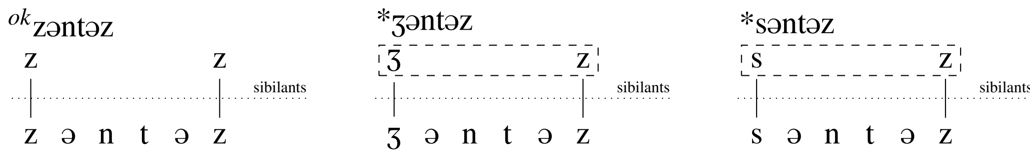


Figure 5: Sibilant harmony in TUAREG

Figure 5 shows that only agreeing sibilants are allowed on the tier, as *zz* in the well-formed word *zəntəz*, whereas combinations that disagree in anteriority (**ʒz*) or voicing (**sz*) are ruled out by the grammar. Both spreadings operate over the same set of elements, therefore as before in the case of double vowel harmonies, one tier is enough to capture the double sibilant harmony in TUAREG.

3.2.2 Imdlawn Tashlhiyt (Berber)

In IMDLAWN TASHLHIYT (Berber), affixal sibilants regressively harmonize with the stem in voicing and anteriority, see (Hansson 2010b, McMullin 2016). Whereas the anteriority harmony is not blocked by anything, the voicing one is blocked by any intervening voiceless obstruents. If there are no sibilants in the stem, the sibilant is realized as [s]. The SPE-style rules generalizing this pattern are given below in (F1-4). The data in (32-41) from (Elmedlaoui 1995, Hansson 2010a) illustrates the harmonic pattern using the causative prefix *s*-.

$$\text{(F1)} \begin{bmatrix} +\text{consonantal} \\ +\text{cont} \end{bmatrix} \rightarrow \begin{bmatrix} \alpha\ \text{anterior} \end{bmatrix} \ / \ \_\_ \begin{bmatrix} +\text{consonantal} \\ +\text{cont} \\ \alpha\ \text{anterior} \end{bmatrix}$$

$$\text{(F2)} \begin{bmatrix} +\text{consonantal} \\ +\text{cont} \end{bmatrix} \rightarrow \begin{bmatrix} \beta\ \text{voice} \end{bmatrix} \ / \ \_\_ \begin{bmatrix} +\text{consonantal} \\ +\text{cont} \\ \beta\ \text{voice} \end{bmatrix}$$

$$\text{(F3)} \begin{bmatrix} +\text{consonantal} \\ +\text{cont} \end{bmatrix} \rightarrow \begin{bmatrix} -\text{voice} \end{bmatrix} \ / \ \_\_ \begin{bmatrix} +\text{consonantal} \\ -\text{sonorant} \\ -\text{voice} \end{bmatrix}$$

$$\text{(F4)} \begin{bmatrix} +\text{consonantal} \\ +\text{cont} \end{bmatrix} \rightarrow [\text{s}] \ / \ \text{elsewhere}$$

In (32), there are no sibilants in the root, and the prefix appears in its by-default, underlying form *s*-. In all other examples, this prefix agrees with the sibilant in a root

(32) s:-uga ‘CAUS-evacuate’
(33) s-as:twa ‘CAUS-settle’
(34) ʃ-fiaʃr ‘CAUS-be.full.of.straw’
(35) z-bruz:a ‘CAUS-crumble’
(36) ʒ-m:ʒdawl ‘CAUS-stumble’
(37) s-ħuz ‘CAUS-annex’
(38) s:-ukz ‘CAUS-recognize’
(39) sˤ-rˤuˤfˤzˤ ‘CAUS-appear.resistant’
(40) s-mχazaj ‘CAUS-loathe.each.other’
(41) ʃ-quʒ:i ‘CAUS-be.dislocated’

in its voicing and anteriority, therefore the possible feature specifications are [–voice, +ant] (33), [–voice, –ant] (34), [+voice, +ant] (35), and [+voice, –ant] (36). However, the anteriority harmony in this language does not have blockers, whereas the voicing spreading is blocked by any intervening voiceless obstruent such as /ħ/, /k/, /f/, /χ/, or /q/. In (37-41), the sibilant in the root is voiced, but the one in the prefix is voiceless because of the intervening voiceless obstruents in-between them.

<table>
<tr><th colspan="2">Sibilant tier
$\mathbf{T}_{ant}$ = {s, z, ʃ, ʒ}</th></tr>
<tr><td rowspan="2">1.</td><td>*[α ant] [β ant]</td></tr>
<tr><td>H_{ant} = {*sʃ, *sʒ, *ʃs, *ʒs, *zʃ, *ʃz, *ʒz, *zʒ}</td></tr>
<tr><th colspan="2">Tier of sibilants and voiceless obstruents
$\mathbf{T}_{voice}$ = {s, z, ʃ, ʒ, ħ, k, f, χ, q}</th></tr>
<tr><td rowspan="2">1.</td><td>*[+ cont, α voice] [+ cont, β voice]</td></tr>
<tr><td>H_{v1} = {*sz, *zs, *ʃz, *ʃʒ, *ʒʃ, *sʒ, *ʒs, *zʃ}</td></tr>
<tr><td rowspan="2">2.</td><td>*[+ cont, + voice] [– sonor, – voice]</td></tr>
<tr><td>H_{v2} = {*zħ, *zk, *zf, *zχ, *zq, *ʒħ, *ʒk, *ʒf, *ʒχ, *ʒq}</td></tr>
</table>

Table 6: TSL grammars for IMDLAWN TASHLHIYT harmony

One tier is not enough, because there is no limit on the number of voiceless obstruents in-between the two sibilants agreeing in anteriority, therefore this process is not local – the locality required for the anteriority harmony cannot be achieved over a single tier that contains both sibilants and voiceless obstruents; see the appendix B explaining this problem in a more formal way. The only solution is to project two tiers, putting only sibilants on one of them (T_{ant}) and blocking the bigrams disagreeing in anteriority (H_{ant}); and projecting both sibilants and voiceless obstruents on the second one (T_{voice}), and blocking sibilants disagreeing in anteriority (H_{v1}) and voiced sibilants followed by voiceless obstruents (H_{v2}).

Figure 6 shows this analysis of the IMDLAWN TASHLHIYT double sibilant har-

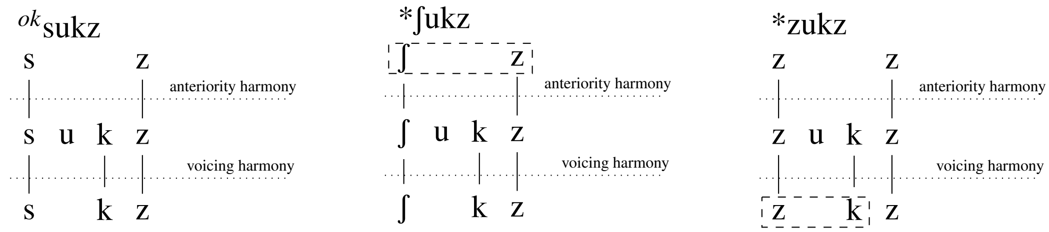


Figure 6: Sibilant harmony in Imdlawn Tashlhiyt

mony. The word *sukz* is well-formed, because the anteriority grammar allows for the *sz* combination: they both agree in anteriority, and the voicing tier is satisfied with the bigrams *sk* and *kz*. However, **ʃukz* is ruled out because the **ʃz* combination is banned over the anteriority tier: note that the sibilants *ʃ* and *z* are not adjacent over the tier of the voicing harmony. The word **zukz* is also out, because the voicing grammar prohibits voiced sibilants followed by the voiceless obstruents (**zk*), and this blocker is not seen over the anteriority harmony tier. Imdlawn Tashlhiyt pattern requires two tiers, because the set of the elements taking part in one long-distance assimilation is different from the one involved in the other spreading: two tiers are required, and their alphabets are in the set-subset relations.

3.2.3 Double sibilant harmonies: summary

In this section, I showed two examples of double sibilant harmonies. For the Tuareg pattern, one TSL grammar was enough, just as it was enough for the double vowel harmonies. But for Imdlawn Tashlhiyt sibilant harmony projecting two tiers is unavoidable, and the tier alphabets of these two grammars are in the set-subset relation.

1. **Tuareg:** all sibilants agree with respect to both voicing and anteriority, so all sibilants are undergoers for both spreadings.

2. **Imdlawn Tashlhiyt:** all sibilants agree in anteriority and voicing, but whereas the voicing harmony can be blocked by intervening voiceless obstruents, the anteriority agreement is unbounded.

3.3 Vowel Harmony and Consonant Harmony

Before, I showed double harmonies only among vowels or only among consonants. Here, the two harmonies target different sets of elements – one of them is consonantal harmony, and the second one is vowel assimilation. In both cases, two tier-based strictly local grammars are needed to capture the pattern, and in both cases the alphabets of these grammars are disjoint, i.e. their intersection is empty.

3.3.1 Kikongo (Bantu)

There are both consonant and vowel harmonies in KIKONGO (Bantu). Vowel harmony enforces vowels to agree in height, whereas nasal agreement turns /d/ and /l/ into /n/ if preceded by a nasal in the stem, see (Ao 1991, Hyman 1998) and the rule in (G1). The data below is adopted from (Hyman 1998). In these examples, the applicative suffix *-el, -il*, and the reversive transitive suffix *-ol, -ul* illustrate the vowel harmony in height.

(G1) $\left[\ \text{– consonantal}\ \right] \rightarrow \left[\ \alpha\ \text{height}\ \right]\ /\ \left[\begin{array}{c}\text{– consonantal}\\ \alpha\ \text{height}\end{array}\right]\ \_\_$

(42)	-somp-el-	'attach-APPL'	(46)	-tomb-ol-	'do-TRANS'
(43)	-leng-el-	'languish-APPL'	(47)	-lemb-ol-	'broom-TRANS'
(44)	-sik-il-	'support-APPL'	(48)	-vil-ul-	'move-TRANS'
(45)	-vur-il-	'surpass-APPL'	(49)	-bub-ul-	'bribe-TRANS'

In this language, suffixes are specified for rounding, and acquire their height specification depending on the stem vowel. In (42, 43) and (46, 47), both vowels in the stem and in the affix are non-high, whereas (44, 45) and (48, 49) contain the high ones.

Vowel tier **T_v = {e, o, i, u}**	
1.	*[α hi] [β hi]
	H_v = {*oi, *ou, *ei, *eu, *io, *ie, *uo, *ue}

Table 7: TSL grammar for KIKONGO vowel harmony

This harmony operates over the tier of vowels T_v, and the grammar must rule out all combinations of vowels that disagree in height. But along with vowel harmony, this language also has a consonantal one – nasal feature spreading formalized in (G2). Segments /d/ and /l/ in the affix both become /n/ if nasal consonants /m/ or /n/ are present in a root. See examples below from (Ao 1991), where *-idi* is the perfective active suffix, and *-ulu* is its passive counterpart.

(G2) /d/, /l/ $\rightarrow$ [n] / [m], [n] __

(50)	-suk-idi-	'wash-PERF.ACT'	(53)	-suk-ulu-	'wash-PERF.PASS'
(51)	-nik-ini-	'ground-PERF.ACT'	(54)	-nik-unu-	'ground-PERF.PASS'
(52)	-meng-ene-	'hate-PERF.ACT'	(55)	-meng-ono-	'hate-PERF.PASS'

In (50, 51), there are no nasals in a root, so the consonant in the affix is unchanged – it remains /d/ and /l/ respectively. However, when there are nasals /n/ or /m/ in the stem, both affixal /d/ and /l/ change to /n/, see (51, 54) and (52, 55) respectively.

$\mathbf{T}_n = \{\mathbf{n, m, d, l}\}$	
1.	*d [+ nasal], *l [+ nasal]
	$H_n = \{$*nd, *nl, *md, *ml$\}$

Table 8: TSL grammar for KIKONGO consonant harmony

Only /d/, /l/, and nasals are involved in the process, therefore those are projected over the tier, and the grammar H_n blocks occurrence of /d/ and /l/ after the nasals. The two tier-based strictly local grammars have absolutely different tier alphabets T_v and T_n, and cannot be combined together, because nasals can occur in-between vowels, as well as vowels in-between nasals.

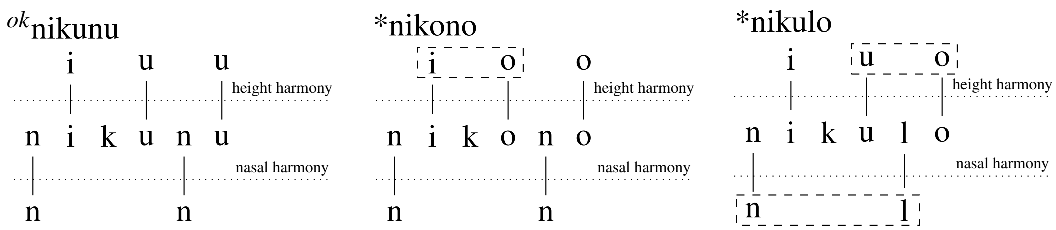


Figure 7: Vowel and nasal harmonies in KIKONGO

Figure 7 illustrates such an analysis for the KIKONGO pattern. Two tiers are necessary for the description of this pattern, because only those can provide the needed locality relations among vowels for vowel harmony, and /d/, /l/ and nasals for the nasal assimilation. The well-formed word *nikunu* is permitted because its vowel tier string *iuu* does not violate the vowel harmony rule, and the nasal tier *nn* also satisfies the nasal harmony. Such ill-formed combinations of segments as **io, *uo*, and **nl* are ruled out by the corresponding grammars. Note that the two vowels /i/ and /u/ are intervening between /n/ and /l/ in the rightmost subfigure, and only existence of the two separate tiers creates locality needed for the TSL analysis of this pattern.

3.3.2 Bukusu (Bantu)

In BUKUSU (Bantu), there are also two harmonies: one applies to vowels, and the other one affects consonants. The system of the vowel harmony is the same as in KIKONGO: all vowels agree in height. The tier alphabet and the set of illicit substrings is also the same: $T_v = \{$e, o, i, u$\}$, and $H_v = \{$*oi, *ou, *ei, *eu, *io, *ie, *uo, *ue$\}$. However, in this case, the consonantal harmony affects liquids instead of nasals: /l/ assimilates to /r/ if it is preceded by /r/, see (H1-2) summarizing the rules of the BUKUSU harmony.

(H1) $\left[-\text{consonantal} \right] \rightarrow \left[\alpha\ \text{height} \right] \ / \ \begin{bmatrix} -\text{consonantal} \\ \alpha\ \text{height} \end{bmatrix} \ \_\_$

(H2) /l/ → [r] / [r] __

This pattern is shown below using the applicative suffix *-el, -il*; cited by (Odden 1994, Hansson 2010a).

(56)	teex-el-	'cook-APPL'	(58)	lim-il-	'cultivate-APPL'
(57)	reeb-er-	'ask-APPL'	(59)	rum-ir-	'send-APPL'

The affixal /l/ is realized as [l] if there is no /r/ in the stem (56, 58), but if preceded by the rhotic liquid, it assimilates to [r] (57, 59).

Liquid tier $\mathbf{T}_{liq}$ = {**l, r**}	
1.	*[+ rhotic] [+ lateral]
	H_{liq} = {*rl}

Table 9: TSL grammar for BUKUSU consonant harmony

Such liquid harmony requires projection of separate tier with the tier alphabet T_{liq} that consists only of liquids, and the grammar H_{liq} blocks a lateral liquid if it is preceded by a rhotic one.

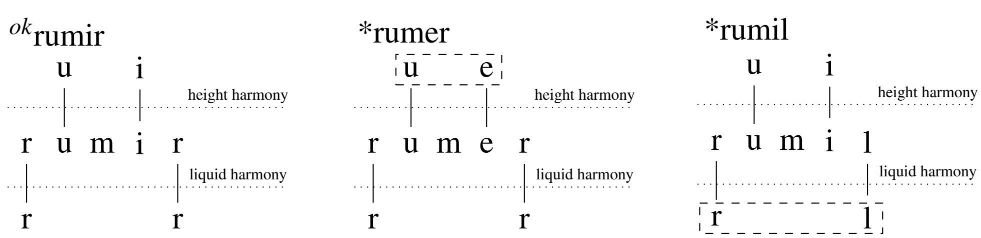


Figure 8: Vowel and liquid harmonies in BUKUSU

Figure 8 shows that this analysis correctly allows the well-formed word *rumir*: both grammars are satisfied with the corresponding tier substrings. However, if **ue* is found on the tier of vowels, H_v bans this word: the two vowels disagree in height. Likewise, H_{liq} prohibits **rumil* because of the illicit combination **rl* found on its liquid tier. Note that both vowels are found in-between the two liquids, that is why only the existence of the two tier allows us to capture both assimilations locally.

3.3.3 Separate vowel and consonant harmonies: summary

In this section, I have shown two examples of separate vowel and consonant harmonies within the same language. In both cases, two TSL grammars were necessary, and the tier alphabets of these grammars are disjoint, i.e. their intersection is empty.

1. **Kikongo:** all vowels agree in voicing (vowel harmony). /d/ and /l/ assimilate to /n/ if preceded by a nasal (consonant harmony).

2. **Bukusu:** all vowels agree in voicing (vowel harmony). /l/ assimilates to /r/ if preceded by a /r/ (consonant harmony).

3.4 Intersecting Tier Alphabets: Non-Attested Example

An example of a harmonic process that requires two TSL grammars, the tier alphabets of which will have non-empty intersection (excluding the proper subset case), would be any pattern in which the same element takes part in two harmonies that operate over a different sets of segments.

For example, imagine a pattern of the non-existent language KIRGHONGO that reminds us of both KIRGHIZ and KIKONGO. Its alphabet includes *a, o, n*, and *d*. Vowels within a word agree in rounding, i.e. it is either /a/ or /o/, unless /n/ intervenes: only non-rounded vowels can follow /n/. The consonant /d/ assimilates to /n/ if preceded by /n/. Obviously, such pattern would require two TSL grammars, where the first one enforces the vowel harmony: $T_v = \{\text{a, o, n}\}$, $H_v = \{\text{*ao, *oa, *no}\}$. The second grammar captures the nasal assimilation: $T_n = \{\text{n, d}\}$, $H_n = \{\text{*nd}\}$. The intersection of the two tier alphabets then contains /n/. However, there are no attested cases like that: if two TSL grammars are needed to capture the double harmony, their tier alphabets are either disjoint, or one is the subset of another.

4 Overview of Double Harmonic Patterns

Below, I summarize the harmonic patterns with spreading of more than one feature in different languages. For each of the languages, I briefly describe the pattern, and also its type in terms of the tier alphabet(s). **S** stands for a single tier – only one TSL grammar is required to capture both spreadings. **E** stands for embedding, i.e. one tier alphabet is a subset of the other one. **D** means that the two tier alphabets are disjoint.

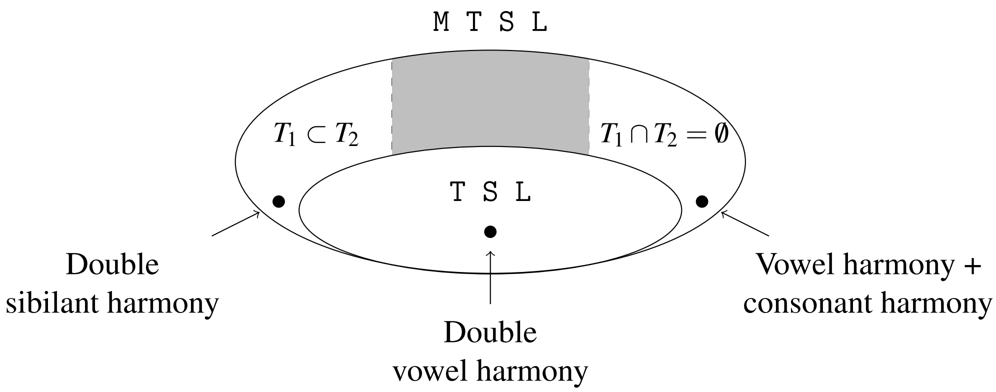
Type	Language	Description	References
S	Kirghiz	All vowels harmonize both in fronting and rounding.	Sec. 3.1.1 Kaun (1995)
S	Kachin Khakass	All vowels agree in fronting. Tier-adjacent high vowels agree in rounding.	Korn (1969)
S	Yakut	All vowels harmonize in fronting. High vowels agree with any preceding vowel, non-high ones agree only with preceding non-high vowel.	Sec. 3.1.3 Sasa (2001)

Type	Language	Description	References
S	Karachay, Azerbaijani, Tuvan, Uyghur	All vowels agree in fronting. High vowels agree with any preceding vowel in rounding.	Kaun (1995) Comrie (1981) Krueger (1977)
S	Kazakh	All vowels agree in fronting. In fronted words, all vowels agree in rounding. In non-fronted ones, only high vowels agree with the preceding vowel in rounding.	Korn (1969)
S	Shor	All vowels agree in fronting. In fronted words, all vowels agree in rounding. In non-fronted ones, tier-adjacent vowels agree in rounding only if they agree in height.	Korn (1969)
S	Buryat, Mongolian	All vowels harmonize in ATR. Non-high vowels harmonize in rounding, high vowels block it. Transparent element for both harmonies is /i/.	Sec. 3.1.2 Poppe (1960) Svantesson et al. (2005)
S	Even, Oroch, Ulcha	All vowels agree in RTR. Tier-adjacent non-high vowels agree in rounding.	Comrie (1981)
S	Tuareg	All sibilants agree both in voicing and anteriority.	Sec. 3.2.1 Hansson (2010b)
E	Imdlawn Tashlhiyt	Sibilants agree in voicing and anteriority. Voiceless obstruents block the voicing harmony and are transparent for the anteriority one.	Sec. 3.2.2 Hansson (2010a)
D	Kikongo, Kiyaka	Vowels agree in height. Consonants /d/ and /l/ assimilate to /n/ if preceded by a nasal.	Sec. 3.3.1 Ao (1991) Hyman (1998)
D	Bukusu	Vowels agree in height. Liquid /l/ assimilates to /r/ if preceded by /r/.	Sec. 3.3.2 Odden (1994)

5 Conclusion

In this paper, I discussed the characterization of harmonies in natural languages that involve more than one feature spreading, in terms of tier-based strictly local grammars. I showed that vowel harmonies tend to fit on the single tier, as was exemplified by KIRGHIZ, BURYAT, and YAKUT: the set of elements involved in each of the spreadings is usually the same, even though their roles – undergoer or blocker – might differ. Consonant (sibilant) double harmonies can also be captured with a single tier (TUAREG), or use another possibility: they might require two TSL grammars with embedding tier alphabets, i.e. the set of items involved in one spreading is a subset of the ones affected by another harmony (IMDLAWN TASHLHIYT). However, there are certain languages with independent vowel and consonantal harmonies, and these two assimilations do not interact with each other (KIKONGO, BUKUSU) – in this case, two TSL grammars are required, and the tier alphabets of these grammars are disjoint, see the figure above, where TSL stands for a single TSL grammar, and MTSL for the local grammar that can project multiple tiers.

This result lays in-between formal language theory and linguistic data, and helps



to bring these two even closer to each other. Kaplan and Kay (1994) concluded that the class of regular languages is enough for the phonology of natural languages, Heinz (2011) reduced this power to the class of the subregular tier-based strictly local languages, and in this paper, I show that for the TSL class, we can accommodate the restrictions on the tier alphabets dictated by the double harmonic systems.

Also, it reveals new typological generalizations about harmonies in general. In line with the discussion of differences in behavior of vowel and consonantal agreement (Walker 2013), and also neurological data showing differences in processing of vowels and consonants (Miceli et al. 2004), I show that whereas most of the double vowel harmonies require just a single TSL grammar, the situation becomes more complicated when the consonantal harmonies are involved.

However, formal characteristics of TSL languages (cf. De Santo (2017) introducing the class of multiple-tier strictly local grammars) do not seem to be different even if their tier alphabet relations are restricted to embedding, intersection, or disjunction. As proved in Appendix C, all of the classes listed above are not closed neither under the operation of intersection, nor under the union, relative complement, or symmetric difference. There are at least two questions to ask. First of all, is there any mathematical or biological reason to avoid learning multiple TSL grammars with intersecting tier alphabets? And, secondly, the linguistic units of phonological descriptions are features – not segments by themselves, therefore what would change if we capture the dependencies by a tier-based strictly local grammars working over the feature tiers, as Graf and Heinz (2016) proposed for syntax? Both of these questions are not answered yet, and solving those might provide us with very useful insights about how the desired universal grammar works.

References

Akinlabi, Akinbiyi. 2009. Neutral vowels in lokaa harmony. *Canadian Journal of Linguistics* 59:197–228.

Ao, Benjamin. 1991. Kikongo nasal harmony and context-sensitive underspecification. *Linguistic Inquiry* 22:193–196.

Chandlee, Jane, Remi Eyraud, and Jeffrey Heinz. 2014. Learning strictly local subsequential functions. *Transactions of the Association for Computational Linguistics* 2:491–503.

Chomsky, Noam. 1965. *Aspects of the theory of syntax*. Cambridge, MA: MIT Press.

Comrie, Bernard. 1981. *Languages of the Soviet Union*. Cambridge, England: Cambridge University Press.

De Santo, Aniello. 2017. Pushing the boundaries of tsl languages. Manuscript. Stony Brook University.

De Santo, Aniello, and Thomas Graf. 2017. Structure sensitive tier projection: Applications and formal properties. Manuscript. Stony Brook University.

Elmedlaoui, Mohamed. 1995. Aspects des représentations phonologiques dans certaines langues chamito-sémitiques. Doctoral dissertation, Université Mohammed V.

Flemming, Edward. 2005. Deriving natural classes in phonology. *Lingua* 115:287–309.

Gainor, Brian, Regine Lai, and Jeffrey Heinz. 2012. Computational characterizations of vowel harmony patterns and pathologies. In *The Proceedings of the 29th West Coast Conference on Formal Linguistics*, 63–71.

Graf, Thomas, and Jeffrey Heinz. 2016. Tier-based strict locality in phonology and syntax. Manuscript. Stony Brook University and University of Delaware.

Hansson, Gunnar Olafur. 2010a. *Consonant Harmony: Long-Distance Interaction in Phonology*. Los Angeles: University of California Press.

Hansson, Gunnar Olafur. 2010b. Long-distance voicing assimilation in berber: spreading and/or agreement? In *Proceedings of the 2010 annual conference of the Canadian Linguistic Association*. Ottawa, Canada: Canadian Linguistic Association.

Heinz, Jeffrey. 2011. Computational phonology part II: Grammars, learning, and the future. *Language and Linguistics Compass* 5:153–168.

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

Herbert, Raymond, and Nicholas Poppe. 1963. *Kirghiz manual*. Uralic and Altaic Series 33. Bloomington: Indiana University Publications.

van der Hulst, Harry, and Noval Smith. 1987. Vowel harmony in khalkha and buriat (east mongolian). *Linguistics in the Netherlands* 81–89.

Hyman, Larry. 1998. Positional prominence and the 'prosodic trough' in yaka. *Phonology* 15:14–75.

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

Katz, Jerrold J., and Paul Postal. 1964. *An integrated theory of linguistic descriptions*. Cambridge, MA: MIT Press.

Kaun, Abigail Rhoades. 1995. The typology of rounding harmony: an optimality theoretic approach. Doctoral dissertation, UCLA.

Korn, David. 1969. Types of labial vowel harmony in the turkic languages. *Anthropological Linguistics* 11:98–106.

Krueger, John. 1977. *Tuvan manual*. Bloomington, Indiana: Indiana University Publications.

Levi, Susannah V. 2001. Tier-based strictly local constraints for phonology. In *CLS 37: The Main Session*, 379–393. Chicago: Chicago Linguistic Society.

McMullin, Kevin James. 2016. Tier-based locality in long-distance phonotactics: learnability and typology. Doctoral dissertation, University of British Columbia.

Miceli, Gabriele, Rita Capasso, Barbara Benvegnu, and Alfonso Caramazza. 2004. The categorical distinction of vowel and consonant representations: evidence from dysgraphia. *Neurocase* 10:109–121.

Nanaev, Kemelbek. 1950. *Uchebnik kirgizskogo yazika*. Frunze: Kirgizgosizdat.

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

Poppe, Nicholas. 1960. *Buriat grammar. Uralic and Altaic series*. Bloomington: Indiana University.

Sasa, Tomomasa. 2001. Yakut vowel harmony: an optimality theory account. *Turkic Languages* 5:270–287.

Sasa, Tomomasa. 2009. Treatment of vowel harmony in optimality theory. Doctoral dissertation, University of Iowa.

Skribnik, Elena. 2003. Buryat. In *The Mongolic Languages*, ed. Juha Janhunen, chapter 5, 102–128. London: Routledge.

Svantesson, Jan-Olof, A. Tsendina, A. Karlsson, and V. Franzen. 2005. *The Phonology of Mongolian*. Oxford: Oxford University Press.

Walker, Rose. 2013. Consonant harmony and vowel harmony: comparisons in typology and sources for nonlocality. Handout. MFM 21, University of Manchester.

Department of Linguistics
Stony Brook University
Stony Brook, NY 11794-4376
alena.aksenova@stonybrook.edu

Appendix

Appendix A introduces the double vowel harmony in TURKISH that can be captured either by two TSL grammars with embedded tiers, or by a single TSL grammar with the length of constraints equal to 3. Crucially, in both cases the structural projection adopted from De Santo (2017) is unavoidable, see the same work and (McMullin 2016) for more examples of the patterns that are impossible to capture using the standard TSL approach. *Appendix B* explains under what configurations one TSL grammar is not enough. *Appendix C* proves that neither of the classes of embedded, intersecting, or disjoint TSL languages is closed under the operations of union, intersection, relative complement, and symmetric difference.

A Structure-Sensitive Projection: Turkish (Turkic)

TURKISH double vowel harmony involves spreading of two features, namely fronting and rounding. All vowels agree in fronting, whereas only high ones agree with any previous vowel in rounding, see the SPE-style rules in (G1-2).

(G1) $\begin{bmatrix} -\text{consonantal} \end{bmatrix} \rightarrow \begin{bmatrix} \alpha \text{ front} \end{bmatrix} \; / \; \begin{bmatrix} -\text{consonantal} \\ \alpha \text{ front} \end{bmatrix} \quad \_\_$

(G2) $\begin{bmatrix} -\text{consonantal} \\ +\text{high} \end{bmatrix} \rightarrow \begin{bmatrix} \alpha \text{ round} \end{bmatrix} \; / \; \begin{bmatrix} -\text{consonantal} \\ \alpha \text{ round} \end{bmatrix} \quad \_\_$

See examples (1-12) from Levi (2001) below, this pattern is exemplified using the genitive suffix with a high vowel (*-in, -ɨn, -un, -ün*), and the plural marker with a non-high one (*-lar, -ler*).

(1)	ip-in	'robe-GEN'	(5)	ip-ʎer	'robe-PL'
(2)	kɯz-ɯn	'girl-GEN'	(6)	kɯz-lar	'girl-PL'
(3)	jüz-ün	'face-GEN'	(7)	jüz-ʎer	'face-PL'
(4)	son-un	'end-GEN'	(8)	son-lar	'end-PL'

(9)	ip-ʎer-in	'robe-PL-GEN'
(10)	kɯz-lar-ɯn	'girl-PL-GEN'
(11)	jüz-ʎer-in	'face-PL-GEN'
(12)	son-lar-ɯn	'end-PL-GEN'

All vowels within a word are either [+front] (cf. the odd numbered examples above) or [-front] (cf. the even numbered ones). The suffix *-lar, -ler* containing a non-high vowel agrees with the stem only in fronting (5-12). The high affix *-in, -ɯn, -un, -ün* agrees with the previous vowel both in fronting and rounding (1-4, 9-12). For example, in (4), the affixal vowel is rounded because it is preceded by /o/, but in (12) the same affix is non-rounded after a non-rounded affixal /e/. So far this system does not differ crucially from those discussed in Sec. 3.1 – a single TSL grammar can capture it.

However, the stem-final /ʎ/ starts its own [+front] harmony domain (see G3). The crucial fact is that it does not affect the labial agreement.

(G3) $\begin{bmatrix} -\text{consonantal} \end{bmatrix} \rightarrow \begin{bmatrix} +\text{front} \end{bmatrix} \; / \; /\text{ʎ}/\# \quad \_\_$

While non-palatalized /l/ is transparent for both vowel agreements (16, 19), its palatalized counterpart /ʎ/ is not: all vowels that follow it, are fronted (17, 20). However, the crucial point is that while being a game breaker for the fronting harmony, it is transparent for the labial one: in (20), the high vowel within the accusative suffix agrees with the rounded stem vowel /u/ even though there is /ʎ/ in-between them.

(13)	kol	‘arm’	(16)	kol-lar	‘arm-PL’
(14)	usuʎ	‘method’	(17)	usuʎ-ʎer	‘method-PL’
(15)	hɯʎaʎ	‘crescent’	(18)	hɯʎaʎ-ʎer	‘crescent-PL’

(19)	kol-u	‘arm-ACC’
(20)	usuʎ-ü	‘method-ACC’
(21)	hɯʎaʎ-i	‘crescent-ACC’

Moreover, only stem-final palatalized /ʎ/ displays such behavior: in other positions it is transparent (18, 21).

This pattern can be captured by a single TSL grammar. However, 2 modifications are needed: first of all, only stem-final /ʎ/ starts its own [+front] harmony domain, whereas it is transparent in all other positions, therefore the procedure of *structural projection* must to be adopted from De Santo (2017) – it allows to project items on the tier depending on their position in the original string. In this case, this position is the end of the stem. Second, to enforce the labial agreement through /ʎ/, the length of the banned sequences must be increased to 3.

Vowel tier + ʎ **T = {a, e, i, ɯ, o, ö, u, ü, ʎ}**	
1.	*[α front] [β front], *ʎ [– front]
	H_f = {*ai, *ae, *aö, *aü, *oi, *oe, *oö, *oü, *ɯi, *ɯe, *ɯö, *ɯü, *ui, *ue, *uö, *uü, *ia, *io, *iɯ, *iu, *ea, *eo, *eɯ, *eu, *öa, *öo, *öɯ, *öu, *üa, *üo, *üɯ, *üu, *ʎo, *ʎu, *ʎa, *ʎɯ}
2.	*[α round] [+ high, β round]
	H_{r1} = {*oɯ, *uɯ, *öi, *üi, *au, *eu, *iu, *ɯu, *aü, *eü, *iü, *ɯü, *öɯ, *üɯ, *oi, *ui}
3.	*V [– high, + round]
	H_{r2} = {*oo, *uo, *ɯo, *ao, *öö, *üö, *iö, *eö, *eo, *io, *aö, *ɯö, *öo, *üo, *oö, *uö}
4.	*[α round] ʎ [+ high, β round]
	H_{r3} = {*oʎɯ, *uʎɯ, *öʎi, *üʎi, *aʎu, *eʎu, *iʎu, *ɯʎu, *aʎü, *eʎü, *iʎü, *ɯʎü, *öʎɯ, *üʎɯ, *oʎi, *uʎi}

Table 10: TSL grammar for TURKISH vowel harmony: v.1

Indeed, the problem of unboundedness does not arise here: there can be only one stem-final /ʎ/ in-between the two vowels that agree in rounding. The tier alphabet T of such TSL grammar includes all vowels and /ʎ/. Combinations of vowels disagreeing in [front] are banned by H_f, as well as non-fronted vowels following /ʎ/. H_{r1} and H_{r2} ban high vowels that disagree in rounding with any previous vowel, and rounded non-high vowels after any other vowel. Finally, H_{r3} enforces the labial assimilation across the stem-final /ʎ/, and blocks rounded non-high vowels following it. The obtained TSL grammar consists of all the restrictions listed above, and, as shown below, operates over a single tier.

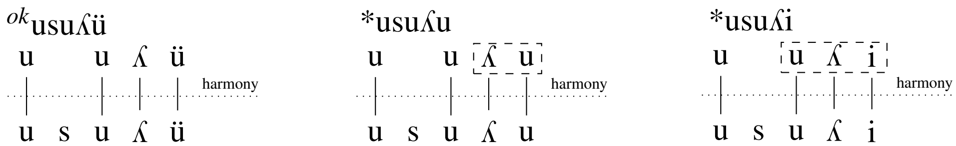


Figure 9: Vowel harmony in TURKISH: first version

Figure 9 shows that the well-formed word *usuʎü* is permitted, whereas its ill-formed counterparts **usuʎu* and **usuʎi* are ruled out because the illicit sequences **ʎu* and **uʎi* are found on the tier. Note that the bigrams *uʎ* and *ʎi* must not be banned on their own, therefore only the enlarged size of the window allows the grammar to correctly rule out **uʎi*, where the two high vowels separated by /ʎ/ disagree in rounding.

Another way to capture this pattern and to avoid using trigrams, is to project two tiers: one for the fronting harmony, and the second one for the labial one.

Tier of vowels + ʎ **T_f = {a, e, i, ɯ, o, ö, u, ü, ʎ}**	
1.	*[α front] [β front], *ʎ [– front]
	H_f = {*ai, *ae, *aö, *aü, *oi, *oe, *oö, *oü, *ɯi, *ɯe, *ɯö, *ɯü, *ui, *ue, *uö, *uü, *ia, *io, *iɯ, *iu, *ea, *eo, *eɯ, *eu, *öa, *öo, *öɯ, *öu, *üa, *üo, *üɯ, *üu, *ʎo, *ʎu, *ʎa, *ʎɯ}
Tier of vowels **T_{lab} = {a, e, i, ɯ, o, ö, u, ü}**	
1.	*[α round] [+ high, β round]
	H_{lab1} = {*oɯ, *uɯ, *öi, *üi, *au, *eu, *iu, *ɯu, *aü, *eü, *iü, *ɯü, *öɯ, *üɯ, *oi, *ui}
2.	*V [– high, + round]
	H_{lab2} = {*oo, *uo, *ɯo, *ao, *öö, *üö, *iö, *eö, *eo, *io, *aö, *ɯö, *öo, *üo, *oö, *uö}

Table 11: TSL grammar for TURKISH vowel harmony: v.2

The first grammar captures the fronting harmony, so its tier alphabet T_f includes all vowels and /ʎ/. It blocks vowels disagreeing in their [front] features, as well as non-fronted segments preceded by /ʎ/. As for the tier of the labial harmony, /ʎ/ is irrelevant, so its tier alphabet T_{lab} does not include them. On this tier, the palatalized lateral cannot intervene between the two vowels agreeing in [round] – it is not projected,

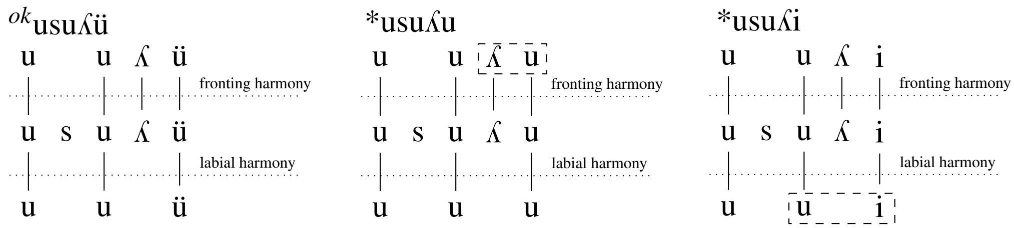


Figure 10: Vowel harmony in TURKISH: second version

therefore H_{lab1} and H_{lab2} rules out the cases of disharmony in a local fashion using bigrams.

Figure 10 shows that word *usuʎü* is well-formed, whereas in other cases, tier substrings **ʎu* and **ui* are banned by the grammars for the fronting and rounding harmonies correspondingly. Such harmonic pattern is the exception from the majority of double vowel harmonies that can be captured with a single TSL grammar, mostly by the necessity of the structural projection: only stem-final /ʎ/ blocks one of the spreadings and does not affect the other one, whereas in other positions it appears to be fully transparent.

B When One Tier Fails: Suffix Substitution

There are certain configurations when double harmony cannot be captured by a single tier. Such cases are important to acknowledge, because they show that the class of strictly local languages with a single tier is not always enough for phonological patterns. Also, the introduction of the multiple tiers raises the complexity of the class from tier-based strictly local (TSL) to multiple tier strictly local (MTSL) languages, see De Santo (2017) for formal details.

For example, consider the IMDLAWN TASHLHIYT pattern discussed above in Sec. 3.2.2. Sibilant harmony in voicing is blocked by any intervening voiceless obstruent, whereas the anteriority agreement among sibilants cannot be blocked. A tier that is blind to the voiceless obstruents will not be able to capture the fact that they are blockers for the voicing harmony, whereas the tier that projects both sibilants and voiceless obstruents will not provide the needed locality relation among sibilants in order to enforce the anteriority agreement that cannot be blocked by those obstruents – their amount in-between the two sibilants is potentially unbounded. For a language to be tier-based strictly local, its tier language must be closed under the operation of *Suffix Substitution*, see (Chandlee et al. 2014).

Definition 1. *(Suffix Substitution Closure)* A language L is strictly n-local iff for all strings u_1, v_1, u_2, v_2, there exists a string x of the length $n-1$ such that if $u_1 \cdot x \cdot v_1$, $u_2 \cdot x \cdot v_2 \in L$, then $u_1 \cdot x \cdot v_2 \in L$.

$$\begin{array}{r|c|ll} \overset{u_1}{\overline{\mathrm{a\,a\,a}}} & \overset{x}{\mathrm{a}} & \overset{v_1}{\overline{\mathrm{b}}} & \in L_1 \\ \overset{u_2}{\overline{\mathrm{a}}} & \mathrm{a} & \overset{v_2}{\overline{\mathrm{b\,b\,b}}} & \in L_1 \\ \hline \overset{u_1}{\overline{\mathrm{a\,a\,a}}} & \mathrm{a} & \overset{v_2}{\overline{\mathrm{b\,b\,b}}} & \in L_1 \end{array} \qquad \begin{array}{r|c|ll} \overset{u_1}{\overline{\mathrm{a\,b\,b}}} & \overset{x}{\mathrm{a}} & \overset{v_1}{\overline{\mathrm{a}}} & \in L_2 \\ \overset{u_2}{\overline{\mathrm{a}}} & \mathrm{a} & \overset{v_2}{\overline{\mathrm{b\,b\,a}}} & \in L_2 \\ \hline \overset{u_1}{\overline{\mathrm{a\,b\,b}}} & \mathrm{a} & \overset{v_2}{\overline{\mathrm{b\,b\,a}}} & \notin L_2 \end{array}$$

Figure 11: Ex. of suffix substitution closure being satisfied (left) and violated (right)

For example, consider the tier language $L_1 = a^*b^*$. Its possible tier sequences are such as *bbbb*, *abbbb*, *aaabb*, or *aa*, whereas **abaa* or **babba* cannot be generated. If this language is tier-based strictly 2-local, the suffix substitution closure predicts that for any string $u_1 \cdot x \cdot v_1$ and $u_2 \cdot x \cdot v_2$ in this language, where x is a substring of the length 1 (i.e. just a single symbol), the string $u_1 \cdot x \cdot v_2$ will also be generated by this language. This entailment holds indeed. Consider the left subfigure above: the strings $aaa \cdot a \cdot b$ and $a \cdot a \cdot bbb$ are included in the language L_1, and the suffix substitution closure correctly predicts the language L_1 to generate the string $aaa \cdot a \cdot bbb$ as well.

As another example, consider the tier language $L_2 = a^*b^*a^*$. Possible strings of this language are *abbbb*, *abbbba*, *bbaaa*, but not **ababb* or **bbabb*. This language is not strictly local: see the right subfigure above. The words $abb \cdot a \cdot a$ and $a \cdot a \cdot bba$ are both well-formed in this language, however, $abb \cdot a \cdot bba$ is not, therefore this language is not strictly local. There is no size of the banned sequence n that will be sufficient to describe this language in terms of n-local constraints, because for every n-locality, we will be able to find a string in this language where $n-1$ a will separate the two b, and the suffix substitution will fail, and therefore this language is not local.

C Brief Overview of Closure Properties

Closure properties characterize classes of languages, more particularly, they define operations and tell the consequences of their application to a language of a certain class: whether the resulting language will still be of the same type as before (closed under some operation), or not (not closed). Apart from the obvious mathematical significance, they also predict new possibilities, and matching those predictions with the linguistic typology can tell us more about the exact formal language classes that corresponds to the class of natural languages.

In this section, I show that TSL languages with restrictions on the tier alphabets (embedded, intersecting, disjoint) are not closed under union, intersection, relative complement, and symmetric difference.

Definition 2. *(Union)* Union ($\cup$) of the sets A and B includes all elements presented in these two sets, $A \cup B = \{x : x \in A \text{ or } x \in B\}$.
Example: If $A = \{a,b\}$ and $B = \{a,c\}$, then $A \cup B = \{a,b,c\}$.

Definition 3. *(Intersection)* Intersection ($\cap$) of the sets A and B includes all elements that are presented in both of these sets, $A \cap B = \{x : x \in A \text{ and } x \in B\}$.
Example: If $A = \{a,b\}$ and $B = \{a,c\}$, then $A \cap B = \{a\}$.

Definition 4. *(Relative complement)* Relative complement ($\setminus$) of the sets A in B includes all elements of A that are not presented in B, $A \setminus B = \{x : x \in A | x \notin B\}$.
Example: If $A = \{a,b\}$ and $B = \{a,c\}$, then $A \setminus B = \{b\}$.

Definition 5. *(Symmetric difference)* Symmetric difference, or the disjunctive union ($\triangle$) of the sets A and B includes all elements in their union except those that are in their intersection, $A \triangle B = (A \cup B) \setminus (A \cap B)$. It is also the union of both relative complements, $A \triangle B = (A \setminus B) \cup (B \setminus A)$.
Example: If $A = \{a,b\}$ and $B = \{a,c\}$, then $A \triangle B = \{b,c\}$.

C.1 Embedded tier alphabets

For each proof in this section, I use ones of the following TSL languages, where the tier alphabet of L_1 and L_3 are supersets of the tier alphabets of L_2 and L_4, correspondingly.

- Language $L_1 = \{a,d\}^*\{b,d\}^*\{c,d\}^*$, and its tier language $T(L_1) = a^*b^*c^*$ over the tier alphabet $T_1 = \{a,b,c\}$. In this language, no c can followed by a or b, and no b can be followed by an a.

- Language $L_2 = \{b,c,d\}^*\{a,b,d\}^*$, and its tier language $T(L_2) = c^*a^*$ over the tier alphabet $T_2 = \{a,c\}$. Here, all c must precede all a.

- Language $L_3 = \{a,c\}^*\{b,c\}^*$, and its tier language $T(L_3) = a^*b^*$ over the tier alphabet $T_1 = \{a,b\}$. All a must precede all b.

- Language $L_4 = \{a,c\}^*b\{a,c\}^*$, and its tier language $T(L_4) = b$ over the tier alphabet $T_2 = \{b\}$. There must be exactly one b in a well-formed string of this language.

Lemma 1. Two TSL languages, if one of their tier alphabets is a subset of the other one, are not closed under the union, intersection, relative complement, and symmetric difference.

Proof. Given the two languages L_1 and L_2, their union $L_1 \cup L_2 = \{a,d\}^*\{b,d\}^*\{c,d\}^* \cup \{b,c,d\}^*\{a,b,d\}^*$. This language is not TSL, because there is no n-gram that will allow to restrict the occurrence of a to either before, or after c, but not both, due to the unbounded amount of c. Such strings as *adcccca* are predicted by the suffix substitution closure, but they are not in the language $L_1 \cup L_2$.

For L_1 and L_2, their intersection $L_1 \cap L_2 = \{a,d\}^*\{b,d\}^* \cup \{b,d\}^*\{c,d\}^*$. The similar problem arises: the amount of b is unbounded, therefore no window of the size n can restrict either all a to appear before b, or all c to follow after b, but not both: the suffix substitution closure predicts such words as *abdbdbbc*, and they are not in the language $L_1 \cap L_2$.

The relative complement of the language L_3 in L_4 is $L_4 \backslash L_3 = \{a,c\}^*bc^*a\{c,a\}^*$. The unboundedness of a does not allow the local grammar to ensure that at least one b is presented in a well-formed string of the language, so such words as *aaacaca* might be generated, and they are not in the language $L_4 \backslash L_3$.

The symmetric difference of languages L_1 and L_2, $L_1 \triangle L_2 = d^*a\{d,a\}^*\{b,d\}^*\{c,d\} \cup \{b,d\}^*c\{c,b,d\}^*\{a,b,d\}^*$. In this language, the amount of c is unbounded, and local grammars cannot restrict the occurrence of a to either before c, or after, but not both, if the amount of c is unbounded. Such words as *acccda* are predicted, and they are not in the language $L_4 \triangle L_3$.

□

C.2 Disjoint tier alphabets

For each proof, I use the following two TSL languages with disjoint tier alphabets as the counterexample:

- Language $L_1 = \{a,c\}^*bc^*$ and its tier language $T(L_1) = a^*b$ over the tier alphabet $T_1 = \{a,b\}$. In this language, a unique b must follow any amount of a.
- Language $L_2 = \{a,b\}^*c\{a,b\}^*$ and its tier language $T(L_2) = c$ over the tier alphabet $T_2 = \{c\}$. There must be exactly one c in a well-formed string of this language.

Lemma 2. TSL languages, tier alphabets of which have an empty intersection, are not closed under the union, intersection, relative complement, or symmetric difference.

Proof. Given the languages L_1 and L_2, their union $L_1 \cup L_2 = \{a,c\}^*bc^* \cup \{a,b\}^*c\{a,b\}^*$. The unbounded amount of a in this language does not allow to restrict either b to be presented once in a string and have unbounded amount of c, or vice versa. Suffix substitution also predicts such words as *aabcaabc* that are not in $L_1 \cup L_2$.

For the same languages L_1 and L_2, their intersection $L_1 \cap L_2 = a^*ca^*b \cup a^*bc$. The unbounded amount of a does not let us to restrict the occurrence of c and b just once in the string in terms of locality: strings like *aabcaabc* that are not in $L_1 \cap L_2$ are also predicted.

The relative complement of L_1 in L_2 is $L_1 \backslash L_2 = \{a,c\}^*bc^+ \cup \{a,c\}^*c\{a,c\}^*bc^+ \cup \{a,c\}^*c\{a,c\}^*c\{a,c\}^*bc^*$. The unboundedness of c does not restrict b to occurrence just once per string. Or, in the other direction, the relative complement of L_2 in L_1 is $L_2 \backslash L_1 = a^*ca^* \cup \{a,b\}^*c\{a,b\}^*b\{a,b\}^*b\{a,b\}^* \cup \{a,b\}^*b\{a,b\}^*c\{a,b\}^*b\{a,b\}^* \cup \{a,b\}^*b\{a,b\}^*b\{a,b\}^*c\{a,b\}^*$. In this case, the unbounded amount of b does not let us to locally ensure that only one c is presented in a string, therefore the language $L_2 \backslash L_1$ is not local.

The symmetric difference of languages L_1 and L_2 is $L_1 \triangle L_2 = a^*b \cup a^*c\{b,a\}^* \cup \{a,b\}^*ca^*b\{a,b\}^*$, where the c must be used just once. However, it is impossible to capture with a TSL grammar, because the amount of a in-between the two c can be unbounded.

□

C.3 Intersecting tier alphabets

For each proof, I use the alphabet of the original string $\Sigma = \{a,b,c\}$ and the following two TSL languages with the intersecting tier alphabets as the counterexample:

- Language $L_1 = \{b,c\}^*ac^*$ and its tier language $T(L_1) = b^*a$ over the tier alphabet $T_1 = \{a,b\}$. It predicts exactly one a after any amount of b.
- Language $L_2 = b^*cb^*ab^*$ and its tier language $T(L_2) = ca$ over the tier alphabet $T_2 = \{a,c\}$. Every well-formed string of this language contains exactly one c and one a after it.

Lemma 3. TSL languages with intersecting tier alphabets are not closed under the operation of union, intersection, relative complement, or symmetric difference.

Proof. Given the two languages L_1 and L_2, their union $L_1 \cup L_2 = \{c,b\}^*ac^* \cup b^*cb^*ab^*$. The absence of the limit on the amount of b makes it impossible to enforce existence of exactly one a, therefore the language $L_1 \cup L_2$ is not TSL.

The intersection of these two languages $L_1 \cap L_2 = b^*cb^*a$. Again, the unboundedness of b does not let us to enforce the existence of one c in every string of the output, so $L_1 \cap L_2$ is not TSL.

The relative complement of L_1 in L_2 is $L_1 \backslash L_2 = \{c,b\}^*c\{c,b\}^*ac^+ \cup b^*ac \cup b^*a$. The reason why this language is not TSL is the same as before. In the other direction, the relative complement of L_2 in L_1 is $L_2 \backslash L_1 = b^*cb^*ab^+$. Again, the unbounded amount of b makes it impossible to enforce a unique c in every string of the output, so this language is not TSL.

The symmetric difference of these two languages $L_1 \triangle L_2 = \{c,b\}^*ac^* \cup b^*cb^*ab^*$. This language is not TSL also because of the unboundedness of b: we cannot enforce the uniqueness of c in every string.

□