# The computational similarity of binding and long-distance consonant dissimilation

Shiori Ikawa & Adam Jardine (Rutgers University)

It has been claimed that syntactic phenomena and phonological phenomena are similar in their computational complexities (Graf, 2018). Computational analyses of phonological phenomena have revealed that a class of transformational patterns called *subsequential class* captures a great deal of segmental processes (Heinz and Lai 2013, Luo 2016, Payne 2017). This leads to the hypothesis that syntactic processes are also subsequential. We give evidence for this hypothesis by studying Binding Conditions A and B as transformations.

Safir (2014) argues that bound anaphors and pronouns originates as the same item called a *D-bound* (Db, henceforth): Db gets *-self* morphology when it is bound locally, and surfaces in a pronominal form when bound non-locally. We will characterize the binding conditions as transformations shown in (1), abstracting away from the derivational details in Safir's analysis as well as the exceptional behaviors of *-self* anaphors in logophoric contexts or PPs.

(1) a. John$_i$ likes Db$_i$ $\mapsto$ John$_i$ likes himself$_i$.
b. John$_i$ says Mary likes Db$_i$ $\mapsto$ John$_i$ says Mary likes him$_i$.

The Db in (1a) acquires *-self* morpheme as it is bound locally while the Db in (1b) surfaces without the *-self* morpheme as it is bound non-locally. Note that Condition B captured in this way is different from the standard Condition B, which does not require the pronoun to be bound.

We can recapture this transformation by representing the c-command relationship in a string called *command-string* (*CS*): Given a node $n$, this is basically a list of c-commanders of $n$ with the ones closer to $n$ being on the left (Graf & Shafiei, 2019).

(2) a. If T is a dependency tree s.t. node *m* has the children $d_1 \cdots, d_i, d, d_{i+1} \cdots, d_n$ where $n \geq 0$, the immediate command-string $ics(d)$ of *d* is the string $d, d_{i+1} \cdots, d_n$.

b. $$cs(n) := \begin{cases} ics(n) & \text{if } n \text{ is the root of } T \\ ics(n) \cdot cs(m) & \text{if } m \text{ is } n\text{'s parent} \end{cases}$$ (Graf & Shafiei, 2019: 208)

Assuming that binding domain is defined by T (G&S), we can characterize Condition A and B as the transformation of the CS including a Db to the CS including either a reflexive anaphor or a pronoun, depending on whether T intervenes between the Db and the binder in the input. The examples in (1a) and (1b) can be represented as (3a) and (3b) respectively.

(3) a. $cs$(*Db*)= ***Db*** *John likes T C* $\mapsto$ ***himself*** *John likes T C*
b. $cs$(*Db*)= ***Db*** *Mary likes T C John says T C* $\mapsto$ ***him*** *Mary likes T C John says T C*

We can then ask: what is the computational complexity of this function? Studies of phonological processes as string functions as in (3) have shown them to be overwhelmingly *subsequential* (Heinz and Lai 2013, Luo 2016, Payne 2017). A subsequential function is one in which any target can only 'look ahead' unboundedly far in at most one direction. For example, in long-distance consonant dissimilation in Latin as shown in (4), the suffix *-alis* surfaces as *-aris* when preceded by /l/, which is blocked by intervening /r/ and non-coronal consonants (e.g. /g/) (Czer 2010). This alternation in the suffix depends on whether /l/ (and blockers) appear to its left, no matter the distance. However, the information to the right of the target is completely irrelevant.

(4) /nav-al̲is/ $\mapsto$ [nav-al̲is] 'naval' /mil̲it-al̲is/ $\mapsto$ [mil̲it-ar̲is] 'military'
/l̲itor̲-al̲is/ $\mapsto$ [l̲itor̲-al̲is] 'of the shore' /l̲eg̲-al̲is/ $\mapsto$ [l̲eg̲-al̲is] 'legal' (Czer 2010)

Defining this transformation with a *logical transduction* makes this explicitly clear. A logical transduction defines the output using properties defined in the logical description of the input

(Courcelle 1994). Following Chandlee and Jardine (to appear), we use a *quantifier-free* (QF) logic over strings which uses monadic predicates representing properties of the elements in the string (e.g., $r(x)$) and *predecessor* and *successor* functions $p(x)$ and $s(x)$ which refer to the elements immediately preceding or following $x$, respectively. For example, we can define exactly when the /l/ in -*alis* appears as [r] in the output with the following monadic predicates.

(5) a. $follows\text{-}l(x) := l(p(x)) \vee (\neg(r(p(x)) \vee non\text{-}coronal(p(x))) \wedge follows\text{-}l(p(x)))$

b. $\mathrm{r}'(x) := \mathrm{r}(x) \vee (\mathrm{l}(x) \wedge follows\text{-}l(x))$

The formula in (5a) recursively defines exactly when an element $x$ follows an /l/ without intervening /r/: either its immediate predecessor is /l/ ($l(p(x))$) or its immediate predecessor is not a blocker (/r/ or non-coronals) and follows an /l/ ($\neg(r(p(x)) \vee non\text{-}coronal(p(x))) \wedge follows\text{-}l(p(x))$). The formula in (5b) then defines exactly when a segment surfaces as [r]: either it was an /r/ in the input ($\mathrm{r}(x)$) or it was an /l/ *and* it followed an /l/ without intervening blockers ($l(x) \wedge follows\text{-}l(x)$)). (This formula describes the alternation in the suffix, but not the general alternation pattern of [l]s and [r]s.)

Chandlee and Jardine (to appear) show that QF predicates that are defined recursively—but only using *either* predecessor *or* successor, never both at the same time—are guaranteed to describe subsequential functions. Restricting access to either the predecessor or successor function captures the subsequential notion that only information in one direction is available to any input target.

Similarly to (5), (6) captures the transformation in (3) with a logical transduction. (6a) recursively defines when an element $x$ locally precedes a binder, and the formulae in (6b) define what surface as reflexives or pronouns: an element $x$ surfaces as a reflexive when it is a D-bound in the input and locally precedes the binder and as a pronoun when it is a Db in the input and does not locally precedes the binder. The formula in (6a) is clearly parallel to (5a): $x$ locally precedes the binder either when its immediate successor is the binder ($binder(s(x))$) or its immediate successor is not a T and locally precedes the binder ($\neg T(s(x)) \wedge locally\text{-}prec\text{-}binder(s(x))$). Crucially, as (6) uses only successor and (recursively-defined) QF predicates, it is subsequential.

(6) a. $locally\text{-}prec\text{-}binder(x) := binder(s(x)) \vee (\neg T(s(x)) \wedge locally\text{-}prec\text{-}binder(s(x)))$

b. $reflexive'(x) := D\text{-}bound(x) \wedge locally\text{-}precedes\text{-}binder(x)$
$pronoun'(x) := D\text{-}bound(x) \wedge \neg locally\text{-}precedes\text{-}binder(x)$

(We abstract away from the requirement that Dbs have to have c-commanding binders, which can be expressed as logical constraints on the domain of the function. Note also that we define the binders as $\phi$-matching DPs with the Db and do not consider the index—see Rogers (1998) for the motivation—and thus confine the discussion to the cases with only one possible binder.)

This definition makes explicit the similarities between the transformation in (3) and consonant dissimilation: Crucially, a Db searches for the binder uni-directionally in the sense that it only looks to the right. It is also unbounded in the sense that it does not care how many non-T elements are in between the Db and the binder.

We have shown that both binding and consonant dissimilation can be captured with quantifier-free logic over strings enriched with recursive predicates. This introduces a hypothesis for syntax: that syntactic transformations are subsequential; it also introduces a logical characterization of this restrictive class of functions that, in future work, can be investigated over tree representations.

**Select references** Graf, T., & Shafiei, N. (2019). C-command dependencies as TSL string constraints. *Proceedings of SCiL*; Chandlee, J. & Jardine, A. (2019). Quantifier-free Least Fixed Point Functions for Phonology. *Proceedings of MoL 2019*; Safir, K. (2014). One true anaphor. *LI*.

**Let Me Stress How Flat Phonology Is: The segmental feature [stress] in Abkhaz**

Samuel Andersson, Yale University, samuel.andersson@yale.edu

**Introduction** Stress is often analyzed with syllables as the stress-bearing unit (Halle and Kenstowicz 1991, Hayes 1995, Kager 1995). I show that if stress is a property of the syllable, the stress system of Abkhaz cannot be accounted for. In Abkhaz (all data from Spruit 1986 and Yanagisawa 2010), even monosyllabic forms may contain many units relevant for stress computation, as many as there are segments. I argue that individual segments in Abkhaz can contrast for the feature [stress] underlyingly. This analysis requires schwa in Abkhaz to be epenthetic, which is well-supported by exceptionless distributional restrictions. My account correctly predicts several independent facts about Abkhaz phonotactics, and unlike syllabic theories of stress, it explains the possible stress patterns found in roots with two, one, and zero underlying vowels. I conclude that there are empirical reasons to favor featural models of stress, in line with recent work questioning the need for hierarchical representations of stress and prosody generally (Scheer 2008, Faust and Ulfsbjorninn 2018, Andersson 2019, Newell 2019).

**Stress** Abkhaz stress (marked by ´ and cued by intensity; Arshba 1979: 7) is computed by Dybo's Rule, described here in terms of a [stress] feature: stress falls on the first [+stress] immediately followed by [-stress], otherwise stress is final (for implementations, see Dybo 1977, Spruit 1986, Trigo 1990, Kathman 1992, Meurer 2009, Vaux 2015, Hao and Andersson 2019). (1) shows a threeway stress contrast on /CaCa/ roots:

(1) Stress on /CaCa/ roots (-+ and -- do not contrast, so I write -+; Spruit 1986: 41-42)

| Stress | Root (UR) | Definite (/á-ʀᴏᴏᴛ/) | Indefinite (/ʀᴏᴏᴛ-k'/) | Translation |
|---|---|---|---|---|
| ++ | /lábá/ | /á-lábá/ [alabá] | /lábá-k'/ [labák'] | 'stick' |
| +- | /k'ába/ | /á-k'ába/ [ak'ába] | /k'ába-k'/ [k'ábak'] | 'shirt' |
| -+ | /laħʷá/ | /á-laħʷá/ [álaħʷa] | /laħʷa-k'/ [laħʷák'] | 'raven' |

Above we see that /á-k'ába/ → [ak'ába] (first [+stress] immediately followed by [-stress]), while /á-lábá/ receives default final stress: [alabá]. Dybo's Rule is successful at accounting for Abkhaz stress: analysis of all definite-indefinite pairs in Yanagisawa (2010) reveals only 9/644 unpredicted stress alternations (98.6% coverage; Andersson forthc.).

**Schwa** This paper assumes some underlyingly vowelless forms in Abkhaz. For these forms, Dybo's Rule works as in (1), but consonants with primary stress surface with epenthesis as [Cə́] (cf. Spruit 1986: 37). I propose the epenthesis rule Ć → Cə́ / _C. This correctly predicts stress-based alternations, as with the 1sɢ prefix /ś/: /ś-nát͡ʃʷʰá/ [s(*ə)nat͡ʃʷʰá] 'my finger' but /ś-la/ [s*(ə́)la] 'my eye.' An alternative analysis would instead have underlying /ə/ in alternating cases (/sə-/), and a rule of unstressed schwa deletion. However, there are at least three independent restrictions on Abkhaz schwa, exceptionless across 6,153 native and borrowed words in Yanagisawa (2010), which follow automatically if all schwas are epenthetic under stress:

(2) Generalizations and how they are explained by schwaless underlying forms

i) There is no initial schwa, since schwa is only epenthesized after consonants:

[áb] 'the father' no *[əb]

ii) In $V_1V_2$ hiatus, $V_2$ is never schwa, since there is no consonant motivating epenthesis:

[jaág] 'bring it!' no *[jaə́g]

iii) With alternating prefixes C~Cə́, only C appears before vowels, since epenthesis only targets Ć / _C: [s-nat͡ʃʷʰá] 'my finger' [sə́-la] 'my eye' [s-áb] 'my father' no *[sə́-ab]

These three independent restrictions must be stipulated if schwa is underlying, but follow without any further assumptions if schwa is derived by epenthesis.

**Stress types** Armed with Dybo's Rule and schwaless URs, we will look at stress contrasts within roots. If stress is a property of the syllable, we predict that the number of possible stress patterns on roots should change as we manipulate the number of syllables. The data below disconfirm this prediction: the exact same typology of stress types is found in all two-consonant roots, whether they have two, one, or zero underlying vowels. (3) shows using surface forms the threeway stress contrast ++, +-, -+ for all four root shapes /CaCa/, /CCa/, /CaC/, and /CC/:

(3) C(V)C(V) stress patterns (numbers after glosses are frequencies in a 644-word corpus)

| Shape | Root (UR) | /á-/ 'the' | /-k'/ 'a' | Gloss | Shape | Root (UR) | /á-/ 'the' | /-k'/ 'a' | Gloss |
|---|---|---|---|---|---|---|---|---|---|
| Ca Ca | /lá bá/ | a-labá | labá-k' | stick (18) | Ca C | /k'á ś/ | a-k'as**ə́** | k'as**ə́**-k' | shawl (11) |
| | /k'á ba/ | a-k'ába | k'ába-k' | shirt (28) | | /k'ʲá t'/ | a-k'ʲát' | k'ʲát'-k' | rod (23) |
| | /la ħʷá/ | á-laħʷa | laħʷá-k' | raven (13) | | /sa ś/ | á-sas | sas**ə́**-k' | guest (15) |
| C Ca | /ṕʰ ʂ̜á/ | a-pʰʂ̜á | pʰʂ̜á-k' | wind (14) | C C | /ʃ́ʷ ʃ́/ | a-ʃʷʃ**ə́** | ʃʷʃ**ə́**-k' | cloud (14) |
| | /t́' ʂ̜a/ | a-t'**ə́**ʂ̜a | t'**ə́**ʂ̜a-k' | hole (15) | | /q́' z/ | a-q'**ə́**z | q'**ə́**z-k' | goose (25) |
| | /b ná/ | á-bna | bná-k' | forest (41) | | /χʲ ʃ́/ | á-χʲʃ | χʲʃ**ə́**-k' | hawk (14) |

(3) shows that the number and type of stress patterns stay constant as we vary the syllable count. For example, -+ roots are always pre-stressing in the definite form ([á-laħʷa], [á-bna], [á-sas], [á-χʲʃ]), and have final root stress in the indefinite form ([laħʷá-k'], [bná-k'], [sas**ə́**-k'], [χʲʃ**ə́**-k']). The number of stress units is thus independent of the number of syllables, contra syllable-based analyses of stress. The correct surface forms follow from applying Dybo's Rule (DR) and epenthesis (EP) to the URs in (3), where individual segments bear [stress] specifications:

(4) Derivations of selected words

| | | | | | | |
|---|---|---|---|---|---|---|
| UR | /á-k'ába/ | /laħʷá-k'/ | /bná-k'/ | /á-t́'ʂ̜a/ | /á-q́'z/ | /q́'z-k'/ |
| DR | ak'ába | laħʷák' | bnák' | at́'ʂ̜a | aq́'z | q́'zk' |
| EP | ---------- | ----------- | --------- | at'ə́ʂ̜a | aq'ə́z | q'ə́zk' |
| SR | [ak'ába] | [laħʷák'] | [bnák'] | [at'ə́ʂ̜a] | [aq'ə́z] | [q'ə́zk'] |
| Gloss | 'the shirt' | 'a raven' | 'a forest' | 'the hole' | 'the goose' | 'a goose' |

**Conclusions** I have argued that Abkhaz stress is best understood in terms of segments rather than syllables: individual segments contrast for a feature [stress], as in /ʃ́ʷʃ́/, /q́'z/, and /χʲʃ́/. This is motivated by the empirical finding that the exact same stress types exist for all biconsonantal roots, independently of the number of underlying vowels. This analysis relies on [ə] as epenthetic, but this is well-motivated independently of stress: underlying /ə/ fails to explain several entirely exceptionless restrictions which follow automatically from an epenthesis analysis. Abkhaz thus provides empirical support for stress as a segmental feature, following recent work on non-hierarchical approaches to prosody.

**References** **Arshba**, Nelli V. (1979). Dinamičeskoe udarenie i redukcija glasnyx v abxazskom jazyke. Tbilisi: Mecniereba. **Dybo**, V. A. (1977). *Zapadnokavkazskaja akcentnaja sistema i problema ee proisxoždenija*. Paper presented at the Konferencija Nostratičeskie jazyki i nostratičeskoe jazykoznanie. **Spruit**, A. (1986). *Abkhaz Studies* (PhD). Leiden University. **Yanagisawa**, T. (2010). *Analytic Dictionary of Abkhaz*. Tokyo: Hitsuji Shobo.

**Gender Resolution Rules: Monotonic mappings over gender hierarchies**

Sedigheh Moradi
Stony Brook University

**Overview** Resolved agreement predicts the predicate agreement with a subject made up of co-ordinated elements. We study the possible forms of this agreement in 2- and 3-gender paradigms (based on data from Corbett 1991) and find that only a small fraction of the a-priori conceivable patterns of gender resolution are used. We propose that the attested patterns of gender resolution are constrained by the mathematical notion of monotonicity (Graf 2019): given a base hierarchy for gender and a parallel order for the equivalent phonological forms, all agreement patterns are monotonic mappings from the base hierarchy to a fixed ordering of the realized forms. What we call a hierarchy is a prominence relation whose motivation either has a semantic explanation or lies outside language, rendering relationships that arise from third factor principles (Chomsky 2005).

**Data** A fair amount of descriptive work is done on resolution rules that determine the form of agreement used with conjoined nouns (Corbett 1988: 259). No prior literature, however, has investigated the reason behind the limited number of attested patterns of resolved gender agreement. The combination of co-ordinated elements in a 2-gender language, for instance, creates a paradigm with 4 cells, as in French below. If one assumes that there cannot be separate genders just for resolved agreement, then there are $2^4 = 16$ different options. Since in resolved agreement the order of the co-ordinated elements does not matter, our options drop to $2^3 = 8$. Yet only two patterns are attested in our sample of seven 2-gender languages (French, Spanish, Latvian, Hindi, Panjubi, Modern Hebrew and Rumanian). The same happens in 3-gender languages: Out of $3^6 = 729$ possibilities, only a few are attested. This is illustrated by the tables below showing the resolved gender for all the possible gender combinations in three languages. (Abbreviations: N: NEUT: neuter; F: FEM: feminine; M: MSC: masculine; R: rational.)

| **SG** | **PL** |
|---|---|
| MSC | MSC |
| FEM | FEM |

*Table 1:* French

| | **MSC** | **FEM** |
|---|---|---|
| **MSC** | M | M |
| **FEM** | M | F |

*Table 2:* Resolution in French

| **SG** | **PL** |
|---|---|
| MSC | MSC |
| FEM | FEM |
| NEUT | NEUT |

*Table 3:* Icelandic

| | **MSC** | **FEM** | **NEUT** |
|---|---|---|---|
| **MSC** | M | N | N |
| **FEM** | N | F | N |
| **NEUT** | N | N | N |

*Table 4:* Resolution in Icelandic

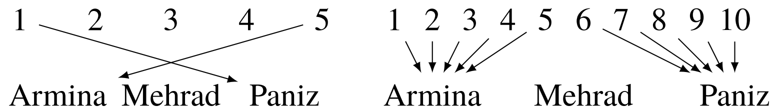
| **SG** | **PL** |
|---|---|
| MSC | RATIONAL |
| FEM | |
| NEUT | NEUT |

*Table 5:* Tamil

| | **MSC** | **FEM** | **NEUT** |
|---|---|---|---|
| **MSC** | R | R | R |
| **FEM** | R | R | R |
| **NEUT** | R | R | N |

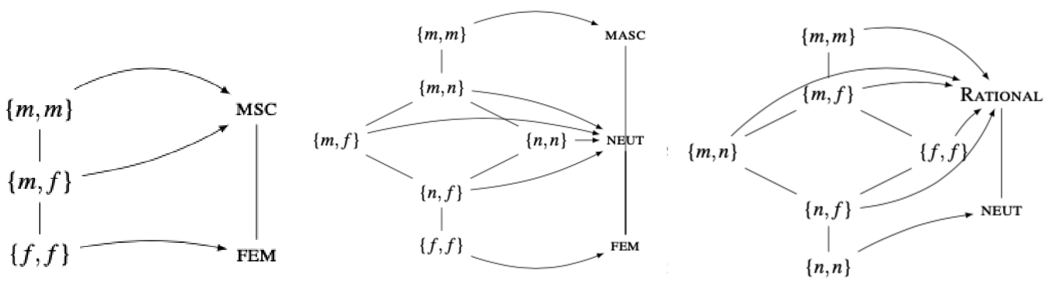
*Table 6:* Resolution in Tamil

**Analysis** For a function that takes conjoined sets as its domain and a hierarchy of gender as its co-domain, all the attested patterns can be accounted for in terms of monotonic mappings. Monotonicity is a mathematical property that corresponds roughly to the linguistic notion of order preservation. If $A$ is a list of ordered numbers and $B$ is a list of names in alphabetical order, then a function $f$ from $A$ to $B$ is monotonic *iff* it preserves the relative order of elements. If $f$ maps 1 to *Paniz* while 5 is mapped to *Armina*, $f$ is not monotonic (this can be seen in crossing branches). However, mapping all the numbers between 1 and 5 to *Armina* and all the numbers from 5 to 10 to *Paniz* still preserves the original order and the function is monotonic.




Monotonicity has already been used as an abstract condition on morphological paradigms to explain typological gaps in adjectival gradation, case syncretism, pronoun syncretism, Person Case Constraint, Gender Case Constraint, and tense stem syncretism (Graf 2019, Moradi 2019).

The data we have sampled so far suggests that gender resolution also has a monotonic backbone. Starting from an underlying hierarchy of, say, $m < n < f$, we can construct a pointwise algebra to represent various gender combinations. Elements of that algebra are then mapped into a hierarchy of plural genders as shown below, and this mapping is indeed monotonic for all the languages in the data set. In most languages, the plural genders are the same hierarchy from which the pointwise algebra is constructed, but some languages like Tamil instead use a condensed hierarchy in this case. Note that the combination of genders is represented as sets instead of tuples because in resolved agreement the order of the co-ordinated elements does not matter.




*(a)* French *(b)* Icelandic *(c)* Tamil

Our analysis of gender resolution mirrors Graf's account of the PCC in that a hierarchy of combinations is constructed from a base hierarchy in a pointwise fashion. In contrast to the PCC, though, an account of resolved gender has to allow for small differences between the base hierarchies used by individual languages: most 3-gender languages use $M < N < F$, but Tamil instead requires $M < F < N$. That said, gender lacks the semantic justification of person, and the factors motivating gender assignment are external to the language (Corbett 1991: 264). The surprising thing, then, is not that gender hierarchies can differ across languages, but the overwhelming uniformity of hierarchies across the available data sample. This suggests that there might be external ordering principles for gender after all, similar to what we see with tense (Moradi 2019). Their precise nature has to be left open for now.

**Conclusion** We showed that a possible explanation for the narrow range of patterns of resolved agreement is the monotonicity principle, based on which the mappings from a fixed underlying hierarchy of gender to output forms preserve the underlying structure of the hierarchy. This establishes a strong upper bound on the range of typological variation. It also allows us to predict impossible resolved agreement rules involving different features (e.g., person and number) with gender as the most puzzling of all (Corbett 1991: 1).

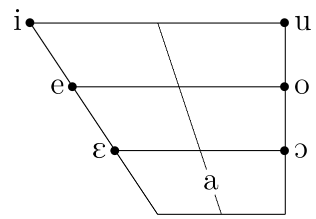
**Corbett, G. 1991**. *Gender*. Cambridge. ● **Corbett, G. 1998**. Resolution Rules in Qafar. *Linguistics 26*: 256-279. ● **Chomsky, N. 2005**. Three factors in language design. *Linguistic Inquiry*, *36*(1): 1-22. ● **Graf, T. 2019**. Monotonicity as an effective theory of morphosyntactic variation. *Journal of Language Modelling* 7 (2), 3-47. ● **Moradi, S. 2019**. *ABA generalizes to monotonicity: an argument from verb stem syncretism. *NELS 49 Proceedings*, v3: 303-315. Amherst: GLSA University of Massachusetts.

# Resolving vowel hiatus in Ikpana

Bertille Baron
Georgetown University

Ikpana (Kwa, Niger-Congo (Ghana); ISO 639-3:[lgq]) is an endangered and largely underdescribed tonal language spoken by approximately 7,500 speakers (Dorvlo 2008). The language counts seven phonemic vowels (see (1)), and three contrastive lexical tones (H, M, L).

(1) Ikpana phonemic vowel inventory




In Ikpana, most nouns and finite verb forms are both vowel-initial and vowel-final (2). Therefore, underlying vowel hiatus is highly commonly found at word boundaries, in a variety of syntactic environments. As the basic word order in the language is SVO (as in (2)), both the SV word boundary and the VO word boundary most commonly display vowel hiatus underlyingly.

(2) a-vá̱ o̱-zá̱ i̱-mbi.
SUBJECT.CM-deer VERB.3SG-cook.PST OBJECT.CM-rice
*'A deer cooked rice.'*

Vowel hiatus and its resolution strategies constitute a widely studied phonological phenomenon. It has been described and analyzed in a variety of languages, and within a variety of theoretical frameworks (i.e. generative, autosegmental, OT, articulatory phonology). Several resolution strategies for vowel hiatus are observed in the world's languages. Casali (1997) lists six different strategies (diphthong formation, vowel elision, glide formation, and coalescence, all involving the reduction of at least one of the two vowels; heterosyllabification and epenthesis, where neither vowel is reduced). Zsiga (1997) also finds that in Igbo, vowel reduction for hiatus resolution purposes is not categorical, but gradient. Various significant factors (i.e. syntactic, lexical, phonological) come into play in deciding which strategy to apply, and also, in cases of resolution by way of reduction, which vowel to reduce (Casali

1997, 2011). These findings reveal that systematic grammatical processes in different modules of grammar can result in phonetic gradience.

The data discussed in this paper is based on a corpus of 1,701 SVO sentences elicited from 5 native speakers of Ikpana (1 female, 4 males; 27–65 years old), for a total of 3,402 tokens of underlying vowel hiatus at word boundaries. In this data set, each one of the 49 possible vowel interactions and 9 possible tonal interactions are represented at the same frequency at each syntactic boundary. All tokens were analyzed in Praat (Boersma 2001), where they were coded for vowel duration, tone duration, F0 (for tone), and F1 and F2 (for vowel quality).

In this paper, I tease apart the effects of three significant conditioning factors affecting vowel hiatus resolution in Ikpana. I show that vowel hiatus at word boundaries in Ikpana is a multiply-conditioned and non-categorical phonological alternation. When it occurs, hiatus resolution is both syntactically and phonologically conditioned. At the syntactic level, preliminary results suggest that identical vowel/tone sequences are handled differently at various word boundaries (i.e. subject-verb vs. verb-object boundary). More specifically, vowel hiatus is more frequently resolved by way of vowel reduction at the verb-object boundary than it is at the subject-verb boundary. At the phonological level, novel findings suggest that hiatus resolution is sensitive to both the interaction of the two underlying vowel qualities, and the interaction of the two underlying tones at a given word boundary. Additionally, the findings support Zsiga's (1997) claim that hiatus resolution is a gradient phenomenon. Not all vowels behave the same way in terms of how much they reduce when they do. Similarly, different underlying tones map differently onto the output TBU(s) at the word boundary when vowel hiatus is resolved.

The contribution of this paper is twofold. Empirically, it adds to the typological description of vowel hiatus resolution based on an example from an endangered and underdocumented language. Theoretically, this paper highlights the crucial role of phonetic methodologies in revealing systematic gradience in the phonological grammar. By applying phonetic methodologies, this work is able to determine both the syntactic and phonological effects determining vowel hiatus resolution, and the gradient effects of vowel reduction for hiatus resolution purposes on specific vowel qualities and tones.

# References


Boersma, Paul. 2001. Praat, a system for doing phonetics by computer. *Glot International* 5(9/10): 341–345.

Casali, Roderic F. 1997. Vowel elision in hiatus contexts: Which vowel goes? *Language*: 493–533.

Casali, Roderic F. 2011. Hiatus resolution. *The Blackwell companion to phonology*: 1–27.

Dorvlo, Kofi. 2008. *A grammar of Logba (Ikpana)*. Doctoral dissertation, University of Leiden.

Zsiga, Elizabeth C. 1997. Features, gestures, and Igbo vowels: An approach to the phonology-phonetics interface. *Language*: 227–274.