Abstract

Stratal Harmonic Serialism: Typological Predictions and Implications

Shira Calamaro

2017

Serialism has played a major role in much of modern phonological theory, from the ordered rules of *Sound Patterns of English* to the ordered levels of Lexical Phonology. With the advent of Optimality Theory, serialism took a back seat to parallel evaluation. With some of the shortcomings of a fully parallel Optimality Theory, there has been much work on reintroducing some aspects of serialism, while maintaining the key benefit of the theory, that it is predictive and not merely descriptive. This dissertation proposes a framework in which serialism is included in two ways, as a component of Gen, as in Harmonic Serialism and as morphological levels, as in Stratal OT, with the goal of exploring the set of typological prediction of the framework and its implications for phonological theory.

Chapter 1 provides an overview of serialism in constraint-based frameworks of phonological theory and its relation to the proposed framework. Chapter 2 is a case study of tonal opacity in Kikerewe, using the proposed framework of Stratal Harmonic Serialism. Chapter 3 provides a typology of predicted grammars in the domain of syllable structure. Chapter 4 investigates the interaction between phonology and morphology in the theory, and the implications. Chapter 5 concludes with a summary of the contributions.

Stratal Harmonic Serialism: Typological Predictions and Implications

A Dissertation
Presented to the Faculty of the Graduate School
of
Yale University
in Candidacy for the Degree of
Doctor of Philosophy

by
Shira Nava Calamaro

Dissertation Director: Claire Bowern

December 2017

Table of Contents

List of Figures

List of Tables

Acknowledgments

I would like to express my sincerest gratitude to my committee for their support on this project over the last several years.

To my advisor, Gaja Jarosz, I am beyond grateful for the years of support, comments on endless drafts, and hours spent discussing minor syllables over skype. I have learned so much from you over the past several years and the completion of this project would not have been possible without such a dedicated and caring teacher.

Ryan Bennett, thank you for your insightful comments and encyclopedic knowledge of the literature. I am so appreciative of your willingness to join my committee and take on the role of advisor.

Claire Bowern, thank you for your guidance and feedback on the committee. Your contributions to linguistics and to the linguistic community have been an inspiration.

John McCarthy, thank you for your invaluable feedback. I first wanted to be a phonologist after reading *A Prosodic Theory of Nonconcatenative Morphology* in an Intro to Linguistics class, and have continued to be inspired by your work since. It was an honor to have you as a member of my committee.

Many thanks to other professors I have worked with at Yale, especially Darya Kavitskaya, for advising my qualifying paper which became the basis for this dissertation.

Thank you to my fellow graduate students for their support and friendship over the years, especially Sabina Matyiku, Alysia Harris, Dolly Goldenberg, and Jen Runds.

Thank you to my parents, Esther and Moshe Calamaro, and siblings Dvorah, Aviva, Yaakov, and Leora for their love and support over the years.

Chapter 1 Serialism in constraint-based frameworks

Since the inception of Optimality Theory (Prince and Smolensky 1993), the issue of serialism has been a lingering area of study. Prince and Smolensky (1993) consider the possibility of a parallel or serial Gen. While the former has been predominant in constraint-based phonological theory, serialism has remained an ongoing area of research. Harmonic Serialism (McCarthy 2000), an OT framework with serial iteration between Gen and Eval, has been used to investigate areas problematic for parallel OT, like locality (McCarthy 2008, 2009, Kimper 2008, 2011; Jesney 2011; Pruitt 2010, 2012). Stratal OT (Bermúdez-Otero 1999; Booij 1997; Kiparsky 2000), a constraint-based variant of Lexical Phonology (Kiparsky 1982; Mohanan 1982, 1986), has similarly investigated areas problematic for parallel OT, such as opacity (Kiparsky 2000; Bermúdez-Otero 1999, 2003; Ito and Mester 2001).The motivation for Stratal Harmonic Serialism comes from the goals of a number of constraint-based frameworks in Optimality Theory, specifically Harmonic Serialism and Stratal Optimality Theory.

Harmonic Serialism and Stratal Optimality Theory each contain a serial component that diverges from a fully parallel Optimality Theory, described as a “full departure from parallelism” and “limited departure from parallelism,” respectively (McCarthy and Cohn 1998). For each of these frameworks, the serial component of the grammar provides a way to solve problems faced by the fully parallel version of Optimality Theory. For Harmonic Serialism, the serial component comes in at the level of Gen, where candidates are limited to those differing from the input by a single change defined by a specified set of operations, and Eval, where single-step candidates continue to be evaluated until they reach convergence. This architecture of the grammar permits the derivation to proceed serially, rather than in parallel, as in Classic OT. One of the many benefits of a serial derivation is the restrictiveness it provides to the framework, limiting the possibility of vast overgeneration of unattested languages, as is endemic to parallel OT (McCarthy 2008, 2009, Kimper 2008, 2011, Pruitt 2010, 2012). The restrictiveness of Harmonic Serialism is an important component of the predictive power of the framework, but is not intended as a general solution for undergeneration problems like opacity. For Stratal OT, the serial component of the architecture is found in the multi-level grammar. This framework developed out of a need to impose different constraint rankings on different levels of the phonology to account for differences due to morphology, allowing it to capture some interactions that cannot be covered under a single constraint ranking (Kiparsky 2000; Bermúdez-Otero 2003; Booij 1997). While the generative power of multiple levels of derivation allows it to capture opaque interactions, the free re-ranking of constraints may potentially be too powerful (Fitzgerald 2002; McCarthy 2007a; Wolf 2012).

Stratal Harmonic Serialism is able to capture the benefits of each of these frameworks to create a typology that solves many of their problems of undergeneration and overgeneration. In this chapter, I summarize some of the main arguments for and against each the major frameworks of parallel Optimality Theory, Harmonic Serialism, and Stratal OT. First, the major undergeneration problem of these frameworks is opacity. Stratal Harmonic Serialism is able to cover cases of interlevel opacity due to the stratal organization of the framework and intralevel opacity due to the Harmonic Serialism levels. Second, the problem of overgeneration in Classic OT and Stratal OT is restricted by the gradualness imposed by Harmonic Serialism, resulting in a sufficiently, but not overly restrictive typology. Each of these arguments plays an important role in the foundation of Stratal Harmonic Serialism.

1.1 Optimality Theory

Optimality Theory (Prince and Smolensky 1993) is a framework with a transformative impact on the study of phonology. The theory uses the interaction of violable constraints to select an optimal output from a set of candidates by evaluating them all in parallel. While earlier work made crucial gains, Optimality Theory gave the field predictive power missing from rule-based frameworks. Given a set of constraints and the architecture of the grammar, Optimality Theory makes clear predictions about the languages that are expected to occur and those that are not expected to occur. Mismatches between the typology and attested patterns point toward a need to alter the constraint set or the architecture of the grammar. In the large body of work on Optimality Theory, some problems have been identified, which have led to proposals for variants of the framework. In this section, I review some of the relevant advantages and disadvantages of the theory.

1.1.1 Advantages of Optimality Theory

An important advantage of Optimality Theory is its ability to make testable predictions about the types of grammars that should be attested. With a finite set of markedness and faithfulness constraints, a factorial typology, or typology of all possible constraint rankings, will generate a set of predicted languages. Sources of overgeneration are those patterns predicted by the typology, but not found in any attested languages, while undergeneration is an attested pattern that cannot be generated by any ranking of the constraint set. This ability to create testable predictions is in contrast to rule-based frameworks, in which most patterns can be accounted for some set of ordered rules, but the set of systems predicted to occur or not occur is not a clear result of those frameworks. Some of these typology problems in Optimality Theory can be solved by a reformulation of representations or additions to the set of universal constraints. However, many problems stem from the architecture of the framework, leading to a number of closely-related frameworks. At its core, the idea of constraint interaction and satisfaction allows us to explore the issue of phonological typology in a way not possible in rule-based frameworks.

One such example of typological predictions and benefits over rule-based proposals is in the domain of syllabification. In their introduction to the theory, Prince and Smolensky (1993) elucidate the inherent flaw in a system of rule-based syllabification, as in the case of Imdlwan Tashlhiyt Berber (Dell and Elmedlaoui 1985). This algorithm follows some rule-based generalizations of core syllabification and adjacent consonant adjunction. While the rule-based algorithm is capable of generalizing the syllabification process, Prince and Smolensky note that it is characterized by a "formal arbitrariness," which allows the attested pattern to be expressed, but would likewise allow a large number of unattested patterns to be described as well. Many of the important phonological notions, such as sonority, are left as commentary, rather than an important component of the algorithm. Similarly, the rules could be altered such that they were acceptable, but generated unattested syllabification patterns. The set of rules has no explanatory power beyond the fact that they can account for the data. In contrast, Optimality Theory provides the explanatory power in the constraint set and the architecture of the grammar.

Since the original proposal of the framework, a large body of research has been generated, providing the analyses of phonological systems and also proposing additional constraints to account for patterns which are not covered by the foundational work. Some of these proposals for additional families of constraints include constraint conjunction (Smolensky 1995), positional faithfulness constraints (Beckman 1997, 1998; Lombardi 1999), output-output faithfulness constraints (Benua 1995, 1997), and targeted constraints (Wilson 2000). There are remaining problems in Optimality Theory that have not been solved by the proposal of new constraints, resulting in proposals for modifications to the architecture of the grammar.

1.1.2 Problems for Optimality Theory

While Optimality Theory offers a number of advantages in phonological theory over its predecessors, there are remaining problems which tend to fall into one of two categories: undergeneration and overgeneration. The problem of undergeneration is perhaps the more significant issue in that there are patterns which parallel Optimality Theory simply cannot generate, specifically some types of opacity. The problem of overgeneration is also a problem because the framework can generate languages which are unattested, exhibiting patterns that are so unlike those found in natural language that they are expected to not be possible. These two problems are discussed in more detail in this section.

1.1.1.1 Undergeneration

The biggest problem of undergeneration in Optimality Theory is opacity. Traditionally, opacity was a phenomenon that could be described in terms of the interaction of ordered rules, as described in Kiparsky (1968, 1973). From the start, the framework has recognized the issue of opacity (Prince and Smolensky 1993) and has been the subject of much work (Baković 2011; Idsardi 2000; McCarthy 1996, 2007a). The completely parallel nature of classic Optimality Theory poses a problem for deriving most of these opaque effects because the evaluation of overapplication and underapplication candidates are rarely the most optimal.

The undergeneration of opacity in Optimality Theory is exemplified by a case of counterbleeding opacity in Bedouin Arabic, which has regular processes of palatalization and vowel deletion, as in the mapping /ħaːkim-iːn/ → [ħaːkʲmiːn] (McCarthy 2007a). While each of processes occur independently, the surface form obscures the reason for palatalization, as the conditioning vowel has been deleted. Within a rule-based framework, this interaction is easily explained by ordering palatalization before vowel deletion.

(1) Counterbleeding opacity with rules

Underlying: /ħaːkim-iːn/

Palatalization: ħaːkʲim-iːn

Vowel deletion: ħaːkʲim-iːn

The gradual application of the rules in the derivation results in an intermediate form, [ħaːkʲim-iːn] which would not be present in any fully parallel model except as a losing candidate. The presence of intermediate forms seems to be a necessary component of the successful evaluation of opacity.

Within a fully parallel constraint-based framework like Optimality Theory, there are no intermediate steps of the derivation. While this allows the grammar to select winners that are globally optimal, opacity is not a problem with a globally optimal solution. Both in cases of overapplication or underapplication opacity, the globally optimal winner would be one that is fully transparent.

The failure of opaque candidates to win over transparent candidates in counterbleeding opacity is shown in tableau (2).

(2) Counterbleeding opacity in classic OT (McCarthy 2007a)

/ħaːkim-iːn/	*iCV	Max	*ki	Id(back)
a. ħaːk^{j}miːn		1		1
b. → ħaːkmiːn		1		L
c. ħaːk^{j}imiːn	W1	L		1
d. ħaːkimiːn	W1	L	W1	L

The independent processes of vowel deletion and palatalization require the rankings *iCV>>Max and *ki>>Id(back), respectively. The desired winner, [ħaːk^{j}miːn], is harmonically bounded by another candidate, [ħaːkmiːn], which does not incur a violation of Id(back). For the winner, this violation of Id(back) is not necessary because the deletion of the high vowel simultaneously removes the violations of *iCV and *ki. Because parallel Optimality Theory does not offer a way to order these two interacting processes, the framework cannot generate these types of opaque interactions.

The problems of opacity in OT have been well-documented with a number of attempts within constraint-based frameworks to solve this problem. Stratal Optimality Theory (Bermúdez-Otero 1999; Kiparsky 2000) adds a component of serialism to the derivation. Sympathy Theory (McCarthy 1999) introduces intermediate forms to the derivation. Optimality Theory with Candidate Chains (McCarthy 2007a) evaluates chains of derivations with the ability to generate many opaque interactions. Because none of these have offered an entirely satisfactory general solution to opacity in constraint-based frameworks, it remains an outstanding problem. As serialism is a crucial component of opaque interactions, the current framework of Stratal Harmonic Serialism introduces intermediate forms in two different ways, both within and between levels.

1.1.1.2 Overgeneration

One aspect of the problem of overgeneration in Optimality Theory is the Too Many Solutions problem, where more repairs for a given marked structure are predicted by the constraint set than are attested in natural language (Pater 1999; Steriade 2001; Lombardi 2001; Blumenfeld 2006). Violations of the marked structure *NC̥ are repaired in a number of ways including nasal substitution, post-nasal voicing, and nasal deletion, but not by epenthesis (Pater 1999). This gap is problematic because it is predicted as a possible repair under the ranking of *N̥C>>Dep. Similarly, the marked structure of word-final voiced obstruents could presumably be repaired via any number of processes such as nasalization, lenition, deletion, or epenthesis, but the only attested repair is devoicing (Steriade 2001).

These cases of Too Many Solutions are a problem for Optimality Theory because a major aim of the framework is to generate typologies based on permutations of the constraint set. When the set of generated typologies does not align with attested languages, aspects of the framework need to be re-examined. A potential solution for this problem is offered by Harmonic Serialism. In the next section, I will outline a number of cases in which parallel Optimality Theory overgenerates, but Harmonic Serialism offers a more restricted typology.

1.2 Harmonic Serialism

In the initial conception of Optimality Theory, Prince and Smolenksy (1993) raise a question on the nature of Gen, whether its iteration with Eval is parallel or serial. If Gen operates in parallel, the set of all possible outputs are considered by Eval at once. If Gen operates serially, a limited set of candidates is considered at once, with the derivation proceeding gradually, requiring a theory of operations on Gen. They consider the former approach, initiating a large body of research in parallel Optimality Theory. In recent years, there has been a renewed interest in the second approach, with a restrictive version of Gen.

1.2.1 Advantages of Harmonic Serialism

The advantages of Harmonic Serialism address both the undergeneration and the overgeneration problem. An important property of Harmonic Serialism is restrictiveness, due to the serial nature of its derivations. This restrictiveness limits the set of outputs in the typology that vastly overpredict pathological systems, as compared to the overgeneration found in parallel OT, as shown in the previous section.

Another advantage of Harmonic Serialism is the ability to account for some types of opaque interactions, which cannot be derived in a strictly parallel grammar. This opacity addresses some of the undergeneration problem of parallel OT, and will be useful in Stratal Harmonic Serialism.

1.2.1.1 Restrictiveness

Restrictiveness in Harmonic Serialism has been argued to be an asset of the framework. This restrictiveness is the result of two primary factors, gradualness and local optimality, which often interact with one another.

1.2.1.1.1 Gradualness

Gradualness is a core property of Harmonic Serialism, in which the generation of candidates is limited to those that differ from the input by one operation, as defined by a theory of operations. A final output can vary from the initial input by multiple operations, but each intermediate form only varies by a single change. One particularly illuminating example of gradualness in Harmonic Serialism is found in cluster reduction as an answer to the Too Many Solutions problem (McCarthy 2008). Given a consonant cluster C_1C_2, attested cases of cluster reduction permit the deletion of the first consonant, but not the second. The gradualness requirement of Harmonic Serialism can predict this asymmetry. This analysis relies on a formulation of the onset-licensing constraint, CodaCond (Goldsmith 1990), which crucially penalizes independent place features in coda position.

(3) CodaCond

Assign one violation mark for every token of Place that is not associated with a segment in the syllable onset.

The path to coda deletion involves a two-step process of Place feature deletion, followed by segment deletion. This is illustrated by a harmonic improvement tableau, showing the harmonic improvement at each step in the derivation of the permissible mapping of /patka/ → [pa.ka], with deletion of C_1.

(4) Harmonic improvement in <pat.ka, paH.ka, pa.ka> (McCarthy 2008)

/patka/	CodaCond	HavePlace	Max[Place]	Max
a. pat.ka	*!			
is less harmonic than				
b. paH.ka		*!	*	
is less harmonic than				
c. pa.ka			*	*

In each step of the derivation, the later step is more harmonic than the previous one. The input in (a) is less harmonic than step (b) due to the violation of CodaCond. Step (b) is less harmonic than step (c) due to the violation of HavePlace.

This derivation is similarly possible in a parallel OT grammar. A crucial point in this argument is the blocking of the second possible mapping, /patka/ → [pa.ta], with the deletion of C_2. This mapping is not possible under a Harmonic Serialism analysis because of the intermediate step [pat.Ha] is less harmonic than the input.

(5) Harmonic bounding of <pat.ka, pat.Ha> (McCarthy 2008)

/patka/	CodaCond	HavePlace	Max[Place]	Max
a. pat.ka	*			
is more harmonic under				
every ranking than				
b. pat.Ha	*	*	*	

In this case, the intermediate step of place feature deletion of C_2 is never harmonically improving, and thus bounded by the input.

Further evidence for this type of solution is that the intermediate step in the deletion of C_1 in a cluster is the existence of derivations terminating with the debuccalisation of C_1.

(6) Ranking for debuccalisation (McCarthy 2008)

/patka/	CodaCond	Max	Max[Place]	HavePlace
a. pat.ka	*!			
is less harmonic than				
b. paH.ka			*	*
is more harmonic than				
c. pa.ka		*	*	

These /patka/ → [paH.ka] type mappings are attested in languages like Arbore (Hayward 1984).

The issue of gradualness is central to Harmonic Serialism, and shows that it can solve many of the problems of overgeneration in parallel OT by restricting the set of possible candidates. As in the case of cluster simplification, Harmonic Serialism offers a solution to the Too Many Solutions problem by restricting the set of possible candidates in Gen through gradualness, as well as through local optimality, which requires intermediate forms to be harmonically improving. Stratal Harmonic Serialism inherits this benefit of Harmonic Serialism. In every level of the grammar, the Too Many Solutions problem is minimized or eliminated, whereas in a Stratal OT grammar with parallel levels the problem would remain.

1.2.1.1.2 Local optimality

Local optimality is a property of Harmonic Serialism grammars as a result of gradualness, in which each gradual step of the grammar must be harmonically improving. This property contrasts with global optimality found by parallel OT grammars. A useful illustration of this is the interaction between Final-C and Coda-Cond in each of these grammars (McCarthy 2006).

The distribution of Final-C and CodaCond violations are in near complementary distribution; Final-C is violated when the candidate ends in a vowel and CodaCond is violated when it ends in an obstruent. When the candidate ends in a sonorant consonant, neither constraint is violated. Such a candidate is a global optimum, minimizing the number of possible Final-C and CodaCond violations. McCarthy (2006) presents a possible input /palasanataka/ and shows the different predictions made by parallel and serial OT.

Parallel OT is capable of finding the global optimum in this case by deleting all segments until the word ends in a sonorant, shown in (7):

(7) Final-C,CodaCond>>Max in parallel OT (McCarthy 2007b)

/palasanataka/	Final-C	CodaCond	Max
a. palasanataka	W1		
b. palasanatak		W1	L1
c. palasanata	W1		L2
d. palasanat		W1	L3
e. palasana	W1		L4
f. → palasan			5

This pattern is quite unusual, as it deletes large portions of words until it finds a candidate that ends on an optimal segment. Given that there are no languages known to exhibit such a pattern, this type of input-output mapping seems undesirable.

In comparison, the gradualness requirement of Harmonic Serialism limits the scope of deletion in the candidate set, thus finding only a local optimum. In each case, the output depends on the relative ranking of the two markedness constraints. If Final-C>>CodaCond, the optimal candidate ends in a consonant rather than a vowel.

(8) Final-C>>CodaCond>>Max in Harmonic Serialism (McCarthy 2007b)

/palasanataka/	Final-C	CodaCond	Max
a. palasanataka	W1	L	L
b. → palasanatak		1	1

If Final-C>>CodaCond, the optimal candidate ends in a consonant rather than a vowel. Crucially, the option to select a candidate ending in a sonorant consonant is not possible because such a candidate has not been created by Gen, nor can the grammar gradually arrive to such a candidate because the next intermediate deletion step [palasanata] is not harmonically improving over the faithful candidate [palasanatak].

Similarly, if CodaCond>>Final-C, the grammar selects the faithful candidate, without Gen ever producing the globally optimal candidate.

(9) Coda-Cond>>Final-C>>Max in Harmonic Serialism (McCarthy 2007b)

/palasanataka/	CodaCond	Final-C	Max
a. → palasanataka		1	
b. palasanatak	W1	L	W1

In each of the Harmonic Serialism grammars, the local optimum is selected rather than the global optimum. This seems to be a desired result because of the unattested patterns that the global optima belong to.

A similar problem of local vs. global optima is found in the realm of spreading, where the all-or-nothing "sour-grapes spreading" is predicted for phenomena like nasal agreement. McCarthy (2009) illustrates this problem with a case from of nasal harmony like that found in Johore Malay.

Under this pattern, nasality spreads rightward until it hits a blocker. When the input does not contain a blocker, spreading can be accounted for in parallel OT with the Agree constraint.

(10) Agree without blocker (McCarthy 2009)

/mawa/	*NasFric	Agree-R[nas]	Ident[nas]
a. mawa		W1	L
b. mãwa		W1	L1
c. mãw̃a		W1	L2
d. → mãw̃ã			3

In this case, the candidate with complete spreading wins because it removes the violation of Agree-R[nas]. However, when the input contains a blocker, here a fricative, parallel OT predicts "sour-grapes spreading."

(11) Agree with blocker (McCarthy 2009)

/mawasa/	*NasFric	Agree-R[nas]	Ident[nas]
a. → mawasa		1	
b. mãwasa		1	W1
c. mãw̃asa		1	W2
d. mãw̃ãsa		1	W3
e. mãw̃ãs̃a	W1	1	W4
f. mãw̃ãs̃ã	W1	L	W5

Because nasality cannot spread to completion, thus removing the Agree violation, the optimal candidate is the one with no spreading at all. While this candidate still has an Agree violation, it does not have any additional Ident violations. The desired winner, [mãw̃ãsa] is harmonically bounded by the faithful candidate. This is another case in which the ability of parallel OT to generate globally optimal candidates creates unattested typological predictions. Interestingly, as McCarthy notes, this is both a case of overgeneration and undergeneration, as it predicts unattested sour-grapes spreading, but also fails to account for Johore Malay nasal agreement.

Harmonic Serialism offers a solution to this problem with a theory of Serial Harmony (McCarthy 2009), using an adapted set of constraints. Under this approach, a Share constraint replaces Agree, which was problematic in parallel OT. Though not included in this summary, Iadditonal constraints nitial and Final are used to govern the directionality of spreading. This analysis of nasal harmony shows that sour-grapes spreading is not predicted by Serial Harmony. The tableau in (12) shows the ranking arguments necessary for spreading without a blocker, with Share[nas] crucially ranked above *NasalGlide, *NasalVowel, and Id[nas].

(12) Ranking data for /mawa/ → [mãw̃ã] (McCarthy 2009)

m|a|w|a	Share[nas]	*NasGli	*NasVow	Id[nas]
a. m|a|w|a		1	2	3
b. →mãw̃ã	W3	L	L	L

The tableau in (13) shows the ranking arguments for sour-grapes spreading, which is incompatible with (12).

(13) Ranking data for /mawara/ → [mawara] (McCarthy 2009)

m|a|w|a|r|a	Share[nas]	*NasGli	*NasVow	Id[nas]
a. →m|a|w|a|r|a	5			
b. mãw̃ã|r|a	L2	W1	W2	W3

No possible ranking exists that would allow both mappings /mawa/ → [mãw̃ã] and /mawara/ → [mawara] with Serial Harmony in Harmonic Serialism.

The case of harmony is another example of the important role of locality in Harmonic Serialism. The unattested sour-grapes spreading phenomenon is a global optimum for constraint violations in parallel OT, while the desired local optimum is possible in Harmonic Serialism. Within a level, the ability to find local optima is inherited in Stratal Harmonic Serialism.

1.2.1.2 Some types of opacity

In addition to restrictiveness, Harmonic Serialism to account for some opaque interactions that cannot be generated in parallel OT. While not a general solution to opacity, some types of opaque interactions can be analyzed in Harmonic Serialism. One such case is that of opaque stress-epenthesis interactions (Elfner 2009). Levantine Arabic usually follows the Latin stress rule (penultimate if heavy, antepenultimate if light), but epenthesis into consonant clusters can create an opaque stress pattern (Abu-Salim 1982; Farwaneh 1995). Elfner (Elfner 2009) identifies two cases of opaque stress that are amenable to the serial interaction of stress and epenthesis in Harmonic Serialism.

(14) Stress-Epenthesis interactions (Elfner 2009)

a. Opaque Pattern 1: Final CC clusters

/katab-t/	(ka)(tá)(bit)	'I wrote'	*(ká)(ta)(bit)

b. Opaque Pattern 2: Medial CCC clusters

/katab-l-ha/	(ka)(tá)(bil)(ha)	'he wrote to her'	*(ka)(ta)(bíl)(ha)

c. Transparent Pattern: Medial CCCC clusters

/katab-t-l-ha/	(ka)(tab)(tíl)(ha)	'I wrote to her'

Elfner shows that the opaque patterns can surface when the operation for stress assignment applies before epenthesis, and transparent when the order is reversed. The gradualness requirement of Gen in Harmonic Serialism is necessary for operations to be ordered in this manner.

The analysis assumes a theory of gradual syllabification prior to stress assignment (Elfner 2009), but for now I will summarize just the details of the stress and epenthesis steps[1]. The first 4 steps of the derivation involve syllabification of the input /katab-t/ and output the intermediate form [(ka)(tab)(t)] as the input to step 5.

(15) /katab-t/: Stress assignment (Elfner 2009)

/(ka)(tab)(t)/	Onset	AlignR (St, Wd)	ParseSeg	Syll-Head	Parseσ	DepV	NoCoda	Foot Bin
a. →(ka)[(táb)](t)				1	2		1	
b. (ka)(tab)(it)	W1			L	W3	W1	W2	
c. (ka)(tab)(ti)		W1		L	W3	W1	1	
d. (ka)(ta)(bt)				1	W3		L	
e. (ka)(tab)(t)				1	W3		1	

While Gen provides epenthesis as a possible candidate in this step, the winner is candidate (a) with stress assignment. Neither epenthesis candidate (b) nor (c) wins due to the ranking of Onset and AlignR(Stem,Word) over Syll-Head. In the following step, resyllabification occurs, removing the violation of NoCoda at the expense of FootBin.

[1] The details of gradual syllabification (Elfner 2009) will be discussed in detail in Chapter 4.

(16) /katab-t/: Resyllabification (Elfner 2009)

/(ka)[(táb)](t)/	Onset	AlignR (St, Wd)	ParseSeg	Syll-Head	Parseσ	DepV	NoCoda	Foot Bin
a. →(ka)[(tá)](bt)				1	2			1
b. (ka)[(táb)](it)	W1			L	2	W1	W2	L
c. (ka)[(táb)](ti)		W1		L	2	W1	W1	L
d. (ka)[(táb)](t)				1	2		W1	L

In the final operation, epenthesis occurs in the minor syllable, removing the violation of Syll-Head at the expense of DepV.

(17) /katab-t/: Epenthesis (Elfner 2009)

/(ka)[(tá)](bt)/	Onset	AlignR (St, Wd)	ParseSeg	Syll-Head	Parseσ	DepV	NoCoda	Foot Bin
a. →(ka)[(tá)](bit)					2	1	1	1
b. (ka)[(tá)](bt)				W1	2	L	L	1

This winner shows opaque stress because the penultimate stressed syllable is no longer heavy, so we would expect transparent stress to appear on the antepenultimate syllable instead. The derivation of opaque stress in medial CCC clusters occurs in a similar manner, with stress assignment preceding epenthesis, while in the case of transparent stress in medial CCCC clusters, epenthesis precedes stress. Due to the gradualness of

Harmonic Serialism, these operations can be ordered, allowing for the possibility of the opaque stress assignment seen in Levantine Arabic.

This case of opacity in Harmonic Serialism demonstrates that the framework is able to handle some types of opaque interactions, particularly those which can be characterized by the interaction of prosodic structure interacting with a phonological process. This ability is important for Stratal Harmonic Serialism because it shows that the framework will be able to handle some cases of opacity within a level, due to this property of gradualness in Harmonic Serialism. In Chapter 6, I will show other similar cases of Stratal Harmonic Serialism capturing this type of structure-building opacity within a single level.

1.2.2 Remaining problems for Harmonic Serialism

While Harmonic Serialism is able to account for some types of opacity, it cannot account for all opaque interactions. The failure of HS to account for many types of opacity led to the proposal of OT-CC (McCarthy 2007), a framework somewhat related to Harmonic Serialism with added mechanisms for the evaluation of entire derivations, or candidate chains. HS fails to account for the same counterbleeding cases as parallel OT, and also does worse in some cases of counterfeeding opacity which parallel OT can account for (McCarthy 2007). Given the failure of Harmonic Serialism as a general solution to opacity, some other mechanism or change to the architecture of the grammar is needed.

1.3 Stratal OT

Stratal Optimality Theory is a version of classic Optimality Theory in which the grammar is separated into multiple levels, or strata, of OT grammars based on the morphology

(Bermúdez-Otero 1999; Kiparsky 2000). This framework is based on Lexical Phonology and Morphology (Kiparsky 1982; Mohanan 1986), with levels using constraint interaction rather than rules, with the need the need for different constraint rankings recognized in early work on Prosodic Morphology (McCarthy and Prince 1993b). The framework has the advantage of being able to account for additional morphological interactions, including opacity, due to the introduction of intermediate forms. However, the framework does have remaining problems of undergeneration and overgeneration.

1.3.1 Advantages of Stratal OT

One of the most transparent benefits of Stratal Optimality Theory is that it introduces some serialism into Optimality Theory. This serialism allows the framework to introduce intermediate forms into the derivation of an output, permitting some types of opacity. A number of analyses have shown how opacity would work in such a framework (Bermúdez-Otero 2003; Collie 2008; Ito and Mester 2001; Kiparsky 2000; Rubach 2000, 2003a, 2003b). I will summarize two such analyses here, one counterfeeding and one counterbleeding.

One such example is a case of counterfeeding in German, as analyzed by Ito and Mester (2001). German has two independent processes, ʀ-vocalization in coda position, /tyːʀ/ → [tyːɐ̯] 'door', and dorsal fricative assimilation after back vowels, /buːç/ → [buːx] 'book.' These processes interact opaquely, as in /dʊʀç/ → [dʊɐ̯ç], *[dʊɐ̯x] 'through,' with the [ç] allophone surfacing, despite its presence before the back vowel, [ɐ̯]. These two processes are in a counterfeeding relationship, with dorsal fricative assimilation ordered before ʀ-vocalization. In the Stratal OT analysis, dorsal fricative assimilation is ordered at the lexical level, while ʀ-vocalization is ordered at the postlexical level. The tableaux of lexical and postlexical levels in (19)-(20) show how this counterfeeding interaction is derived in Stratal OT for the opaque form [dʊɐ̯ç] (Ito and Mester 2001).

The process that applies at the lexical level is dorsal fricative assimilation, which occurs due to the ranking of VEL>>Id(back), giving mappings like /buːç/ → [buːx], as shown in (18).

(18) German counterfeeding: Lexical (Ito and Mester 2001)

/buːç/	VEL	Id(back)
a. buːç	*!	
b. → buːx		*

While dorsal fricative assimilation does occur at the lexical level, ʀ-vocalization crucially does not apply at the lexical level, given by the ranking *ɐ̯ >> *Coda/ʀ. The failure of ʀ-vocalization to apply at the lexical level is shown in (19).

(19) German counterfeeding: Lexical (Ito and Mester 2001)

/dʊʀç/	*ɐ̯	*Coda/ʀ	VEL	Id(back)
a. → dʊʀç		*		
b. dʊʀx		*		*!
c. dʊɐ̯ç	*!		*	
d. dʊɐ̯x	*!			*

At the postlexical level, the relevant processes are reversed by a change in the constraint rankings. Dorsal fricative assimilation no longer applies, but ʀ-vocalization does apply.

(20) German counterfeeding: Postlexical (Ito and Mester 2001)

/dʊʀç/	*Coda/ʀ	*ɐ̯	Id(back)	VEL
a. dʊʀç	*!			
b. dʊʀx	*!		*	
c. → dʊɐ̯ç		*		*
d. dʊɐ̯x		*	*!	

In (20), there is a new ranking of *Coda/ʀ >>*ɐ̯, which allows ʀ-vocalization in the mapping /dʊʀç/ → [dʊɐ̯ç]. Dorsal fricative assimilation can no longer apply at the postlexical level, due to the new ranking of Id(back)>>VEL. This analysis prevents transparent candidates like *[dʊɐ̯x] from winning, leaving the correct surface form [dʊɐ̯ç] with underapplication opacity.

In the case of German counterfeeding opacity, the processes that interact opaquely can be separated into different levels, adding the ability to have intermediate steps in the derivation. A similar tactic is used in counterbleeding opacity. Canadian Raising exhibits a case of counterbleeding opacity that has been discussed extensively (Joos 1942; Chomsky and Halle 1968). This case of counterbleeding opacity has been analyzed in Stratal OT, with the interacting processes of raising and flapping occurring at different levels of the grammar (Bermúdez-Otero 2003).

Canadian English exhibits this classic problem of opacity. The first process is the raising of the diphthongs /aɪ/ and /ɑʊ/ before voiceless obstruents.

(21)		Diphthong raising (Bermúdez-Otero 2003)			
	a.	nəif	‘knife’	naɪvz	‘knives’
		hʌʊs	‘house’	hɑʊzɪz	‘houses’
	b.	ləifər	‘lifer’	laɪ fər mi	‘lie for me’

The data in (21)a show the underlying low diphthongs in the plural forms, but the raised diphthongs in the singular forms ending in voiceless obstruents. The data in (21)b show that diphthong raising fails to apply across word boundaries, as it applies in forms like [ləifər] ‘lifer’, but not in [laɪ fər mi] ‘lie for me’. This provides evidence that vowel raising cannot be a phrase-level process.

The second relevant process is the flapping of obstruents /t/ and /d/ to [ɾ], which is shown in (22).

(22)		Flapping (Bermúdez-Otero 2003)			
	a.	fæt	‘fat’	fæɾər	‘fatter’
		mæd	‘mad’	mæɾər	‘madder’
	b.	hɪt	‘hit’	hi hɪɾ æ	‘he hit Ann’
		hɪd	‘hid’	hi hɪɾ æ	‘he hid Ann’

Flapping occurs within words, as seen in the minimal pairs in (22). Additionally, the application of flapping across word boundaries, as seen in (22), indicates that flapping must occur at the phrase level.

These two processes interact because flapping creates a voiced obstruent, which would bleed the environment for diphthong raising to occur. Raising is ordered before flapping, so these two processes are in a counterfeeding relationship, with overapplication opacity occurring in surface forms.

(23) Counterbleeding

	‘writing’	‘riding’
UR	/raɪt-ɪŋ/	/raɪd-ɪŋ/
Raising	rəitɪŋ	raɪdɪŋ
Flapping	rəiɾɪŋ	raɪɾɪŋ

The data in (23) show the outputs of this counterbleeding opacity, with overapplication occurring in forms like [rəiɾɪŋ] ‘writing’ due to a raised diphthong in the unexpected environment of a voiced obstruent. The Stratal OT account places the two processes at different levels of the grammar: diphthong raising at the stem level and flapping at the phrase level (Bermúdez-Otero 2003).

In the case of German counterfeeding (Ito and Mester 2001) and English counterbleeding (Bermúdez-Otero 2003), opacity is handled by ordering processes serially at different levels of the grammar.

1.3.2 Problems for Stratal OT

The ability to have intermediate forms and re-rank constraints between levels increases the generative power of Stratal OT over a strictly parallel Optimality Theory. However, this framework still has remaining problems of undergeneration and overgeneration.

1.3.2.1 Undergeneration

While Stratal OT does offer a solution for those cases of opacity which are interlevel, due to some morphological or phrasal effect, this does not provide comprehensive coverage of the range of opaque interactions. There are cases where opacity occurs within a single level of the grammar. While Stratal OT can account for many cases of opacity, it does not provide a general solution to the problem. While this should not rule out the importance of the ability to separate morphological effects by level, it does not solve the problem of opacity in constraint-based frameworks.

Let us consider one such case in which opacity occurs within a single level. McCarthy (2007) cites the case of Catalan, in which nasal place assimilation and cluster simplification interact opaquely, as in the mapping /bɛn-k/ → [bɛŋ] (Kiparsky 1985). Under a traditional Stratal OT analysis, the process of nasal assimilation would need to ordered in an earlier level than that of cluster reduction for this counterfeeding interaction. However, phrases like [pɔ.n | ən.tik], *[pɔn.t | ən.tik] 'old bridge' show that cluster reduction must be lexical because it cannot be bled by resyllabification at the phrase level. If place assimilation precedes cluster simplification it must also be lexical, thus leaving Stratal OT unable to account for this opaque interaction.

1.3.2.2 Overgeneration

A major criticism of Stratal OT is its seeming capacity for overgeneration, with the unrestricted constraint rerankings permitted between levels. While it seems that there is indeed a potential for overgeneration in Stratal OT, as there is in parallel OT, there are not an overwhelming number of studies which spell out precisely the problems of overgeneration predicted in a Stratal framework. A notable exception is in Wolf (2012), which deals with cases of stress inversion, though it is not clear that such a problem would carry over to a version with Harmonic Serialism levels. While there have been some attempts at restricting the amount of reranking between levels (Kiparsky 1997, Koontz-Garboden 2003), none have been entirely consistent between analyses. There is no consensus among proponents of Stratal OT regarding what constraints on rerankings, if any, would be needed to restrict the theory.

While the criticisms for the potential overgeneration in Stratal OT are certainly valid, there is still potential for a theory of restriction on rerankings in the framework. However, the claims of overgeneration from constraint reranking may be somewhat exaggerated, as I discuss in chapters 3 and 4. This is particularly true in the reranking found in a serial framework like Stratal Harmonic Serialism. The gradualness of Harmonic Serialism restricts any large scale changes to the phonological system, especially with regard to prosodic structure.

1.4 Stratal Harmonic Serialism

In this dissertation, I argue for a novel framework based on the two frameworks of Harmonic Serialism and Stratal OT. Each of these two frameworks has its own problems of overgeneration and undergeneration for the typology, as discussed in this chapter. Stratal Harmonic Serialism improves on many of these typological problems through the interaction of the two component frameworks. In this section, I outline the arguments for Stratal Harmonic Serialism with respect to undergeneration and overgeneration, which serve as the motivation for the framework and will be investigated thoroughly in the remaining chapters.

1.4.1 Opacity

For both Harmonic Serialism and Stratal OT, the major source of undergeneration is opacity. While the problem of opacity has been problematic for the strictly parallel version of classic OT, each of these two frameworks can account for some additional cases of opacity by introducing a serial component to the grammar. The proposal of serial constraint-based frameworks refers back to the original rule-based formulations of opacity, which were, for the most part, unproblematic. Later attempts to solve opacity in constraint-based frameworks without the use of serialism have been largely unsuccessful. In the case of Stratal OT and Harmonic Serialism, neither provides a general solution to the problem of opacity, though each is able to handle a specific type of opacity. The need for intermediate derivational forms in each of these cases suggests that some degree of serialism is needed to account for most opaque interactions. However, neither framework alone introduces a sufficient amount of serialism to fully solve this problem.

In Stratal OT, opaque interactions are due to morphological interactions between levels of the grammar, crucially depending on the ability to rerank constraints between levels. The serialism in the grammar comes from the intermediate output forms that are found at each level of the grammar. Stratal OT is best able to account for opaque interactions when constraint rankings that correlate with different phonological process occur at distinct levels of the grammar. A constraint ranking that allows a process at an earlier level may no longer be available at a later level when the environment for it is created, resulting in a counterfeeding interaction, as seen in the interaction between dorsal fricative assimilation and ʀ-vocalization in German (Ito and Mester 2001). Similarly, a process may fail to apply at an early level, but the constraint ranking will allow it to apply at a later level, resulting in counterbleeding, as seen in the interaction between flapping and diphthong raising in Canadian English (Bermúdez-Otero 2003). While any counterfeeding or counterbleeding account could theoretically be applied to Stratal OT, there are a number of cases in which there is evidence that the two interacting processes must apply at the same level. In these cases, there must be an intra-level account of the opacity.

In the case of Harmonic Serialism, opaque interactions are possible when they are due to the serial interaction of some phonological process with prosodic structure. This was seen in the case of opaque stress-epenthesis interactions in Levantine Arabic, where prosodic structure is built gradually, with an intermediate step of epenthesis (Elfner 2009). The ability of Harmonic Serialism to capture this type of opaque generalizations crucially depends on the serial building of structure, such as syllabification. Earlier works in Harmonic Serialism make some assumptions about automatic syllabification and resyllabification, which allow syllabification to apply in a single operation along with other phonological processes. The theory of syllabification as a free operation is not compatible with this view of structure building opacity, which uses gradualness to its advantages in eliminating globally optimal or transparent candidates that can be generated with syllabification. Following this lead of opacity in Harmonic Serialism through the serial interaction of structural phonological processes, I propose that many more types of opacity can be accounted for in a single level in Harmonic Serialism. One case that has been problematic is counterbleeding opacity in Catalan, in which it has been argued that cluster simplification and nasal place assimilation must occur in a single level. These two processes must be ordered with nasal place assimilation preceding cluster simplification because cluster simplification would otherwise bleed the environment for nasal place assimilation, as shown in (24).

(24) Counterbleeding in Catalan (Kiparsky 1985)

Underlying: /bɛn-k/

Nasal place assimilation: [bɛŋk]

Cluster simplification: [bɛŋ]

Surface: [bɛŋ]

The evidence for Catalan counterbleeding as a case of intralevel opacity comes from the occurrence of resyllabification across word boundaries in forms like /pɔnt ən.tik/ → [pɔ.nən.tik] 'old bridge.' In this case, cluster simplification could have been resolved through resyllabification, but is not because the /t/ must have been deleted before the phrase level (McCarthy 2007). This has been taken as evidence against the Stratal OT model because not all cases of opacity can be characterized as interlevel, and the machinery of the parallel OT levels cannot account for this counterbleeding as intralevel opacity. However, this case is rescued by Stratal Harmonic Serialism, if Harmonic Serialism can generate this counterbleeding opacity in a single level using a form of gradual cluster simplification, as in McCarthy (2008). Without gradual cluster simplification, this case would still be problematic for Harmonic Serialism, with the transparent candidate [bɛn] repairing markedness constraints against cluster and place assimilation in a single operation of deletion. The desired intermediate candidate with nasal assimilation [bɛŋk] would be eliminated by the fell-swoop candidate [bɛn] under any ranking that permitted both operations.

In this case, the structure building of the syllable and gradual deletion of the second consonant in the cluster allow this to derive the opaque candidate by eliminating the fell-swoop candidate [bɛn], which would repair both offending markedness constraints of *Complex and Share[place], through a revised theory of possible operations. Without the fell-swoop candidate, the derivation proceeds through a set of intermediate associations of place features and gradual deletion of the velar stop. This case provides an example of counterbleeding opacity in Harmonic Serialism through phonological structure and gradual derivations. It furthers the claim that both the Harmonic Serialism and the Stratal OT components of the framework contribute to the overall success of the model.

Stratal Harmonic Serialism improves on both of these frameworks by handling both types of opacity, phonological and morphological, rather than just a single type. The framework makes a prediction about the types of opaque interactions that will be attested; namely, that they will fall into the class of either phonological or morphological opacity. The morphologically opaque interactions can be accounted for through the stratal component of the grammar, with interacting processes occurring on different levels through different constraint rankings. Those cases which cannot be attributed to morphological opacity, and therefore must occur within a single level, are predicted to be due to a serial interaction with phonological structure. The success of this framework to account for the typology of opaque interactions will be covered in Chapter 6.

1.4.2 Levels

Strictly parallel evaluation in classic OT has the problem of overgeneration due to the existence of undesired globally optimal candidates, as seen in the pathologies summarized in the previous section. While Harmonic Serialism restricts the problem of overgeneration, Stratal OT compounds this problem, by permitting multiple levels of potentially overgenerating parallel grammars, with constraint rerankings permitted between these levels. Stratal Harmonic Serialism hypothesizes that the restrictiveness of Harmonic Serialism is sufficient in limiting the powerfulness of Stratal OT, when they interact in a single framework.

Harmonic Serialism is a more restrictive framework than parallel OT, limiting the problem of overgeneration through its properties of gradualness and locality. The version of Gen used in Harmonic Serialism allows only a limited set of candidates, rather than the unrestricted set permitted in parallel OT. This restricts the set of possible winners to those which can be derived through a series of harmonically improving steps, resulting in locally optimal rather than globally optimal winners. The use of levels comprised of serial OT grammars, rather than parallel OT grammars, will reduce the powerfulness of a stratal framework immensely. This restrictiveness on the powerfulness on Stratal OT is trivial given the extensive literature on the restrictiveness of Harmonic Serialism in general. The more interesting question is the extent to which the restrictiveness of Harmonic Serialism can affect the interaction of multiple levels of reranked constraints.

Stratal Harmonic Serialism is a variant of Harmonic Serialism with multiple levels, as in Stratal OT. In this dissertation, I propose that the interaction between these two components makes the framework sufficiently, but not overly, restrictive, due to two major factors. First, the structure building nature of Harmonic Serialism restricts the amount of structure changing that can occur between levels. While constraints can be reranked between levels, gradualness and locality limit the extent to which large scale changes in existing structure can occur. Second, those changes which are possible between levels are neutralized at the final level. Processes which can occur in a locally optimal manner will apply at the final level, neutralizing the distinction due to different rankings at different levels. In Chapter 4, I outline the syllable structure typology predicted by Stratal Harmonic Serialism.

These two generalizations about Stratal Harmonic Serialism will not eliminate all cases of asymmetries due to constraint rerankings, nor should they. There are attested cases of this type, in which there are asymmetries between the syllable types generated at different levels of the grammar, so the architecture of the grammar should not eliminate them completely. In Chapter 5, I present a number of cases of syllable asymmetries that are attested. One such case is found in Donceto Italian (Cardinaletti and Repetti 2009), which shows a difference between levels of the grammar in its treatment of complex clusters. Complex clusters are permitted in morphologically simple words, but consonantal clitics that create complex clusters trigger epenthesis, as shown in (25).

(25) Syllable asymmetries in Donceto Italian (Cardinaletti and Repetti 2009)

a. tri	'three'	ust	'August'
b. ət-rõ:f	'you:sg snore'	pas-ət	'do you:sg pass?'
*t-rõ:f		*pas-t	

The forms in (a) show that complex clusters are permissible initially and finally in surface forms, when they occur in stems. The forms in (b) show the result when consonantal clitics are affixed to verbs, the resulting complex clusters are simplified via epenthesis. The restriction on permissible syllable types differs between levels of the grammar, with the same syllabification pattern predicted if these contrastive forms were analyzed within a single level. Instead, the solution offered by Stratal Harmonic Serialism is a different ranking between the constraints governing syllabification in the word and phrase levels. This example is one case of syllable type asymmetries that provide evidence for the use of distinct levels.

1.4.3 The combined framework

The evidence in the previous two sections shows the independent benefits of having each component of Harmonic Serialism and levels within a single framework. Even more compelling is the case in which both the need to account for opacity and different levels are found in a single language. Spanish offers one such case, with both intralevel opacity and morphology-based syllabification. The evidence for Harmonic Serialism in Spanish is an opaque interaction between s-aspiration and resyllabification that occurs within a single level. The evidence for Stratal levels in Spanish comes from the asymmetry in syllable types from resyllabification across word boundaries. I will preview both of these arguments here, with the extensive analyses provided in Chapter 3 on Syllable Structure Typology and Chapter 4.3 on Opacity.

Evidence for the Harmonic Serialism component comes from the intralevel opaque interaction between s-aspiration and resyllabification in varieties of Caribbean Spanish (Kaisse 1997, 1998). In this data, s-aspiration occurs in coda position. Additionally, word-final codas resyllabify to the onset of the following word if it begins in a vowel. The counterbleeding interaction results in mappings of the form /mes asul/ → [me.ha.sul], with s-aspiration applying, followed by resyllabification. While this type of interaction has traditionally been problematic, if resyllabification occurs gradually, Eval can directly compare candidates in which there is s-aspiration or an intermediate step of resyllabification, but no fell-swoop candidate. This is illustrated in the tableaux shown in (26)-(28). In (26), the possible candidates in Gen are (a) the faithful candidate, (b) the s-aspiration candidate, and (c) the intermediate resyllabification candidate. The winner is the s-aspiration candidate. The following two passes show both gradual steps of resyllabification.

(26) Step 1: S-aspiration

/ [(mes)][(a)(sul)]/	ParseSeg	*s/C	Onset	*Ambisyl	IdSyl	NoCoda
a. [(mes)][(a)(sul)]		W1	1			2
b. →[(meh)][(a)(sul)]			1			2
c. [(me(s)][a)(sul)]		W1	L	W1		2

(27) Step 2: Linking

/ [(meh)][(a)(sul)]/	ParseSeg	*s/C	Onset	*Ambisyl	IdSyl	NoCoda
a. [(meh)][(a)(sul)]			W1	L	L	2
b. →[(me(h)][a)(sul)]				1	1	2

(28) Step 3: Delinking

/ [(me(h)][a)(sul)]/	ParseSeg	*s/C	Onset	*Ambisyl	IdSyl	NoCoda
a. [(me(h)][a)(sul)]				W1	L	W2
b. →[(me)][(ha)(sul)]					1	1

With gradual syllabification, Harmonic Serialism can account for this counterbleeding interaction. However, with automatic syllabification, the fell-swoop candidate will harmonically bound the desired opaque winner, as is seen in failed accounts of opacity in Harmonic Serialism. Revisiting these types of cases will show the specific conditions needed to account for opaque interactions in Harmonic Serialism.

The evidence for the Stratal component comes from a case of syllable asymmetries between levels found in Spanish. At the word level, Spanish syllabification prefers onset maximization over codas, as seen in words like [a.βlaɾ]. At the phrase level, Spanish permits resyllabification across word boundaries, but the preference is for codas over complex onsets, resulting in resyllabification in [klu.β | e.le.ɣan.te], but not in [kluβ. | lin.do]. The different syllabification preferences between the word and phrase level require different constraint rankings, thus this is good morphological evidence for the need for levels and constraint reranking.

The need for both the Harmonic Serialism and the Stratal component within a single language provides support for the framework of Stratal Harmonic Serialism. Neither individual framework alone would be able to account for both patterns described here. Further evidence for both components is provided in the remaining chapters.

1.4.4 Dissertation overview

The framework of Stratal Harmonic Serialism is predicted to improve on the typological problems of two important serial constraint-based frameworks, Harmonic Serialism and Stratal OT, through the interaction of these components. In this chapter, I have laid out the crucial arguments for these predictions, based on prior work in Optimality Theory and its variants. The remainder of this dissertation will provide the details on how these predictions bear out.

In chapter 2, I provide a case study of tonal opacity in Kikerewe, demonstrating a full Stratal Harmonic Serialism analysis. This chapter shows how the framework can capture the complex opaque and morphological interactions that have failed in other frameworks by making use of both different levels of analysis and gradual derivation.

Chapter 3 provides a syllable structure typology in Stratal Harmonic Serialism. In Chapter 3, I evaluate the restrictiveness of the framework by calculating a typology in the domain of syllable structure. This chapter provides evidence that the framework is not overly powerful, for a number of reasons. First, claims of stratal levels overgenerative capacity are exaggerated, as I show in the context of Stratal HS. While the framework does contain some degree of overgeneration, it is not as problematic as some would claim. The final level of the grammar neutralizes small changes that occur between rerankings of the constraints. Second, the use of Harmonic Serialism, rather than parallel OT levels, greatly reduces the powerfulness of the framework. The gradual building of prosodic structure limits the degree to which large scale changes can occur when the constraints are reranked.

In Chapter 4, I investigate the morphological effects predicted by the framework, specifically in the areas of cumulativity, asymmetry, and opacity. For cumulativity, here a kind of additive markedness, I discuss cases from the typology. For asymmetry, here syllabification patterns dependent on morphology, I provide attested cases of asymmetries in syllable types to support the claim that the framework does not have a problem of overgeneration. While Stratal Harmonic Serialism does exhibit some degree of distinction between levels, these examples show that these cases are in fact attested, as in the case of Spanish resyllabification, a desired property of the framework. For opacity, I provide a typology of opaque interactions in Stratal Harmonic Serialism. In this chapter, I show how both components of the framework are crucial in accounting for the range of opaque effects that are attested. The Harmonic Serialism component accounts for those cases of structurally-based opaque interactions, as shown here by the examples of

Levantine Arabic (Elfner 2009) and Caribbean Spanish (Kaisse 1997, 1998). The Stratal component accounts for those cases of morphologically-based opacity, as shown here by the case of counterfeeding in German (Ito and Mester 2001) and counterbleeding in Canadian English (Bermúdez-Otero 2003). This chapter shows the coverage of the framework when the two components are combined.

In Chapter 5, I provide a summary of the dissertation and identify areas for future investigation.

Chapter 2 Tonal Opacity in Kikerewe

Kikerewe is a tonal Bantu language, which has been shown to have a number of complex opaque interactions (Odden 1995, 2000, 2008). This chapter explores tonal opacity in Kikerewe, which provides an interesting case study for the proposed framework of Stratal Harmonic Serialism due to the derivational mechanisms both within and between levels.

Stratal Harmonic Serialism is a constraint-based framework with two serial components. The first component is stratal levels, morphological levels with potentially different constraint rankings as in Stratal OT (Booij 1996, 1997; Bermúdez-Otero 1999; Kiparsky 2000, 20), an OT-type framework based on Lexical Phonology (Kiparsky 1982; Mohanan 1986). The second component is Harmonic Serialism (Prince and Smolensky 1993; Black 1993; McCarthy 2000) levels, where the Gen component of each level iterates serially with Eval.

This chapter is organized into 3 sections: 2.1 provides the data on Kikerewe, 2.2 details an analysis in Stratal Harmonic Serialism, and 2.3 gives concluding remarks.

2.1 Kikerewe Tonology

This section gives an outline of the facts of Kikerewe. Section 2.1.1 provides an introduction to the processes which occur and section 2.1.2 gives an overview of the interaction of these processes.

2.1.1 Processes

This section describes the basic processes found in Kikerewe, as described by Odden (1995, 2000, 2008): tone spread, Meeussen's rule, tone shift, syllabic fusion, and lapse avoidance. I largely follow Odden's generalizations about the data, though the analysis in section 2.2 will diverge considerably.

Before I present the data, it is important to note some assumptions about Kikerewe tonology. Kikerewe has one phonological tone, which is high (Odden 2000, 2008). The default tone is low. Tone bearing units (TBUs) in Kikerewe are the syllable. TBUs which are not associated with a high tone surface as low. These assumptions will be relevant to the processes described here.

This section presents the processes primarily in terms of surface tone patterns. Section 2.2 will develop an analysis that relies on autosegmental representations and explains their interactions.

2.1.1.1 Tone spread

Tone spread is a process which causes an underlying high tone to appear with an additional high tone on the following tone bearing unit (TBU) in the surface form. In the following form, the high tone on the initial syllable spreads to the second syllable:

(29) /bá-ku-baziila/ → [bákúbaziila] 'they who are sewing'

In this example, there is a high tone on the initial syllable underlying, but it appears on both the initial and the following syllable in the surface form.

The evidence for tone spread can be seen in the following minimal pairs:

(30)	a. ku-bala	'to count'
	b. ku-tú-bála	'to count us'
(31)	a. a-bá-kú-lu:nduma	'they who are growling'
	b. a-bá-lú:nduma	'they who growl'

The verb in (30)a, /bala/ is toneless. However, when this same verb is preceded by a high-toned prefix in (30)b, the verb also appears with a high tone. The second surface high tone is due to tone spread from the prefix /tú/. The verb in (31)a, / lu:nduma/ is also toneless because the high tone only spread one syllable to the right, surfacing on the prefix /ku/. In (31)b, there is no intervening prefix, and thus the high tone surfaces on the verb.

An exception to the tone spread rule occurs at the end of a phrase. When a high tone appears on the penultimate syllable, it does not spread to the final TBU. In the following data, regular spreading occurs in (32), but does not occur in (33):

(32)	a. ku-bala	'to count'
	b. ku-tú-bála	'to count us'
(33)	a. ku-sya	'to grind'
	b. ku-tú-sya	'to grind us'

The verb in (32) is disyllabic, so the high tone on the *tú* morpheme can spread to a non-final syllable. However, the verb in (33) is monosyllabic, so the high tone from the *tú* morpheme does not spread.

The final exception to the tone spread rule is the single case in which phrase final high tones spread leftward. This occurs when monosyllabic verbs are high toned. An example is shown in (34):

(34)	a. kú–há	'to give'
	b. ku-há Búlemo	'to give Bulemo'

The high tone on the verb *há* is underlying, and spreads leftward onto the underlyingly toneless *ku* morpheme in (a). Leftward spreading only happens at the end of a phrase, which is seen by the rightward spreading of the phrase medial high tone in (b).

This process of tone spread interacts with a number of other tonal processes in Kikerewe. Tone spread occurs in nearly all surface forms, thus obscuring other tonal processes which occur. To preview the analysis, tone spread will be analyzed as a late phonological process, often obscuring earlier processes.

2.1.1.2 Meeussen's rule

Meeussen's Rule (MR) is an OCP-like rule that deletes the second in a pair of high tones on consecutive TBUs. This rule is found in the phonology of many Bantu languages, such as Tonga (Meeussen 1963). The rule was originally proposed by Meeussen (Meeussen 1963) and later formulated by Goldsmith (Goldsmith 1984), shown in (35):

(35) H → ∅ / H___

This rule can be interpreted as the deletion of a high tone when it is immediately preceded by another high tone.

In Kikerewe, the concatenation of multiple monosyllabic high-toned morphemes creates the input for MR. In morphologically complex words, when these morphemes are concatenated adjacently, this can lead to long sequences of high tones, as in the following underlying forms:

(36) /abatá:-tú-gí-kú-hé:le:zye/ → [abatá:túgikuhe:le:zye] 'they who didn't give it to us for you (remote)'

(37) / abatá:-tú-gí-kála:ngi:zye/ → [abatá:túgikala:ngi:zye] 'they who didn't fry it for us (remote)'

While the first underlying form has five high toned syllables, the surface form has only two. This surface form results from the interaction of tone spread and MR. The surface form has an additional high tone on the fourth syllable, which is the result of tone spread, as discussed in section 3.1.1.1. While it appears to leave the first two high tones undeleted, this is really an interaction between MR and tone spread, with tone spread causing the remaining high hone left by MR to spread to the next syllable.

2.1.1.3 Tone shift

Vowel hiatus is avoided in Kikerewe, though onsetless syllables do occur word-initially. However, they behave differently from other syllables. In Kikerewe, onsetless syllables are not TBUs, so underlying tones on onsetless syllables shift one syllable to the right. Compare the following forms (outputs shown with spreading):

(38)	a. bá-kú-baziːla	'they who are sewing'
	b. a-kú-báziːla	'he who is sewing'
(39)	a. bá-ká-luːnduma	'if they growl'
	b. a-ká-lúːnduma	'if he growls'

The first form, [bákúbaziila], is derived from the underlying form/bá-ku-baziːla/, with the high tone remaining on the initial syllable. In contrast, the second form, [akúbáziila], is derived from the underlying form /á-ku-baziːla/. Here the tone has shifted from the first syllable, which lacks an onset, to the second syllable. These tones then spread to the third syllable via the tone spread process described in 2.1.1.1.

2.1.1.4 Syllabic fusion

Onsetless syllables in Kikerewe are restricted word-medially. The existence of consecutive vowels can occur in underlying forms as a result of onsetless morphemes. However, these vowels undergo a process of syllabic fusion, as seen in the following examples (Odden 1995):

(40)	a. ebi-toːke	‘bananas’	/ebi-ála/ → ebj-áːla	‘fingers’[2]
	b. omu-títma	‘heart’	/omu-aga/ → omw-aːga	‘compulsion’
	c. olu-bíbo	‘fish trap’	/olu-íle/ → olw-íːle	‘sky’

In each line in (40), the two nouns share the same prefix. While the first stem begins with the consonant, the second begins with a vowel. When a vowel final prefix attaches to a vowel initial stem, they undergo a process of syllabic fusion. This is accompanied by compensatory lengthening and gliding.

This process is relevant to tonal interactions in Kikerewe because it fuses two syllables together, effectively removing a syllable. By removing a syllable, syllabic fusion can cause two high tones to become adjacent, creating the environment for MR to apply.

2.1.1.5 Lapse avoidance

Lapse avoidance is a morphologically-determined process which applies when a toneless modifier follows a noun. In this environment, a high tone is inserted on the last TBU of the noun. This inserted tone then spreads onto the first TBU of the modifier via the regular process of tone spreading. Consider the following forms:

[2] The single tone here is due to the fact that spreading does not apply to phrase final TBUs, but it does not have an effect on the opaque processes described in the analysis.

(41) a. oluguhyo 'broken pot'

b. lu:kiza:no 'green (Cl.11)'

c. oluguhyó lú:kiza:no 'green broken pot'

(42) a. ekikálá:ngilo 'frying pan'

b. kizito 'heavy'

c. ekikálá:ngiló kízito 'heavy frying pan'

While the underlying forms of the nouns do not have a high tone on the final syllable, the high tones appear in noun-modifier phrases, and then undergo the regular process of tone spread. This is shown below:

(43) /oluguhyo luukizaano/ → [oluguhyó lúúkizaano] 'green broken pot'

The process of lapse avoidance is restricted to nouns before toneless modifiers. Odden (2008) showed that this tone insertion rule fails to apply when the modifier contains a tone anywhere in the input:

(44) [oluguhyo luzímá] 'good pot'

In this form the modifier [luzímá] already contains a high tone, and no tone in inserted on the final syllable of the noun. Thus the presence of a tone in the second word of the phrase blocks the tone insertion process, though spreading of that tone still applies.

2.1.2 Interactions

Here I will introduce the interactions between these processes, with a full Stratal Harmonic Serialism analysis given in 2.2. The interactions in this section rely on specific autosegmental representations of high tones. The high tones are marked with an acute accent, as well as with subscripts to represent their autosegmental representations. The differences between these representations violate different constraints, which will be explained in further detail in the analysis.

(45) Different high tone representations for [táká]

a. Singly-linked high tones: [$\text{tá}_1\text{ká}_2$]

```
H H
| |
ta ka
```

b. Doubly-linked high tone: [$\text{tá}_1\text{ká}_1$]

```
  H
 / \
ta  ka
```

The subscripts in these forms will be used as shorthand for the two different autosegmental representations. Subscripts indicate the syllable to which a high tone is associated. The form in (2a) has two singly-linked consecutive high tones with each distinct high tone, indicated by a different numerical subscript. The form in (2b) has a single doubly-linked high tone, indicating by the same subscript on two different syllables. The subscript notations are used as shorthand to clarify the autosegmental representation being used.

2.1.2.1 MR interacts with tone spread

MR interacts with tone spread because on the surface it is ambiguous as to which process is responsible for which high tone. Consider the following form:

(46) /abatá$_1$:-tú$_2$-gí$_3$-kú$_4$-hé$_5$:le:zye/ → [abatá$_1$:tú$_1$gikuhe:le:zye] 'they who didn't give it to us for you (remote)'

While the underlying form has five high tones, the surface form has two high tones. These two surface high tones are the result of the remaining high tone left after MR being spread to the following syllable with the creation of a new association line. Only high tones remaining after MR undergoes tone spread.

The derivation of this form occurs in multiple steps, with MR applying first to delete the consecutive high tones, followed by tone spread to associate singly linked high tones. These steps are shown in (47):

(47)	/abatá$_1$:-tú$_2$-gí$_3$-kú$_4$-hé$_5$:le:zye/	Underlying form
	abatá$_1$:tú$_2$gí$_3$kú$_4$he:le:zye	MR
	abatá$_1$:tú$_2$gí$_3$kuhe:le:zye	MR
	abatá$_1$:tú$_2$gikuhe:le:zye	MR
	abatá$_1$:tugikuhe:le:zye	MR
	abatá$_1$:tú$_1$gikuhe:le:zye	Tone spread
	[abatá$_1$:tú$_1$gikuhe:le:zye]	Surface form

In this derivation, MR applies first, deleting all consecutive high tones. When MR can no longer apply, tone spread applies to the remaining singly linked high tone. Without the autosegmental tonal representations, this interaction is obscured.

2.1.2.2 Tone shift counterfeeds MR

In the following data, the surface forms contain consecutive high tones due to the application of tone shift:

(48) /á$_1$-ku-chú$_2$mita/ → [akú$_1$chú$_2$mí$_2$ta] 'he who is stabbing'

The consecutive high tones in this form are the result of onsetless tone shift, which shifts the high tone to a TBU immediately preceding another high tone. The environment for consecutive tone deletion to apply is present, but it does not occur. This underapplication opacity is due to the ordering of MR and tone shift. In the following derivation, MR is ordered first, followed by tone shift:

(49)	/á$_1$-ku-chú$_2$mita/	Underlying form
	á$_1$kuchú$_2$mita	MR (No application)
	akú$_1$chú$_2$mita	Tone shift
	akú$_1$chú$_2$mí$_2$ta	Tone spread
	[akú$_1$chú$_2$mí$_2$ta]	Surface form

MR is ordered first, but cannot apply because there are no consecutive high tones. In the next step, tone shift creates the environment for MR to apply. In this form, tone shift counterfeeds MR because tone shift would feed MR if the processes had been ordered differently. The effect of tone spread is also seen in the surface form, which has three consecutive high tones, though it is not opaque.

2.1.2.3 Syllabic fusion counterfeeds MR

In this form, there are also consecutive high tones in the surface form:

(50) /a-kí$_1$-a-ká$_2$laːnga/ → [achá$_1$ːká$_2$lá$_2$ːnga] 'he is still frying'

The consecutive high tones in this form are the result of syllabic fusion, in which an onsetless syllable in between two high toned TBUs fuses with the preceding syllable. This is another case of underapplication opacity, due to the consecutive high tones in the surface form that would provide the environment for MR to apply. The following derivation shows that this opacity is due to the ordering of MR and syllabic fusion:

(51) /a-kí$_1$-a-ká$_2$la:nga/ Underlying form

akí$_1$aká$_2$la:nga MR (No application)

achá$_1$:ká$_2$la:nga Syllabic fusion

achá$_1$:ká$_2$lá$_2$:nga Tone spread

[achá$_1$:ká$_2$lá$_2$:nga] Surface form

The process of MR is ordered first but cannot apply. Syllabic fusion follows, creating consecutive high tones which cannot be deleted because MR has already applied. This is a case of syllabic fusion counterfeeding MR. As in the previous example, the effect of tone spread is seen again in the third high tone.

2.1.2.4 Lapse avoidance feeds tone spread

Lapse avoidance is a process which inserts a high tone on the final syllable of a noun in a noun-modifier phrase, if the modifier is toneless. In this form a high tone is inserted, which then spreads on the following TBU:

(52) /kubala kwa:ko/ → [kubalá$_1$ kwá$_1$:ko][3]

Lapse avoidance feeds tone spread, providing the environment for tone spread to apply by inserting a singly linked high tone, shown in the following derivation:

[3] At this stage I am glossing over the details of how lapse avoidance inserts a high tone. This will be explained in greater detail in the analysis section.

(53)	/kubala kwa:ko/	Underlying form
	kubalá$_1$ kwa:ko	Lapse avoidance
	kubalá$_1$ kwá$_1$:ko	Tone spread
	[kubalá$_1$ kwá$_1$:ko]	Surface form

Lapse avoidance must be followed by tone spread because a singly linked high tone is inserted, but a doubly linked high tone appears in the surface form.

2.1.2.5 Tone spread bleeds lapse avoidance

In this form, a noun modifier phrase, lapse avoidance does not occur:

(54) /ku:mbá$_1$la kwa:ko/[4] → [ku:mbá$_1$lá$_1$ kwa:ko]

Lapse avoidance is a process which inserts a high tone on the final syllable of a noun in a noun-modifier phrase, if the modifier is toneless. In this case, there is a noun-modifier phrase and the modifier is toneless, but the lapse avoidance process of tone insertion does not apply, as seen by the lack of spreading to the initial syllable of the modifier.

(55)	/ku:mbá$_1$la kwa:ko/	Underlying form
	ku:mbá$_1$lá$_1$ kwa:ko	Tone spread
	ku:mbá$_1$lá$_1$ kwa:ko	Lapse avoidance (No application)
	[ku:mbá$_1$lá$_1$ kwa:ko]	Surface form

In this derivation, lapse avoidance does not apply because tone spread blocks the TBU onto which the high tone would be inserted. If lapse avoidance were ordered first, it would be able to apply.

[4] This is a simplified version of the underlying form, which would have an additional step of tone shift from the /ḿ/ morpheme.

While in this form tone spread must precede lapse avoidance, in the previous case tone spread must follow lapse avoidance. To derive both [kubalá$_1$ kwá$_1$ːko] and [kuːmbá$_1$lá$_1$ kwaːko], tone spread will precede and follow lapse avoidance. In a rule-based framework, this would require multiple applications of the tone spread rule. However, in Stratal Harmonic Serialism this can be accomplished with a single ranking, as will be shown at in the phrase level section of the analysis.

2.2 Stratal Harmonic Serialism Analysis

In this section, I provide an analysis of the Kikerewe data in Stratal Harmonic Serialism. Section 2.2.1 provides details on possible steps in Gen. Section 2.2.2 provides an outline if the analysis. Sections 2.2.2-5 provide the analysis at each of the three levels of the grammar: stem, word, and phrase.

2.2.1 Steps in Gen

In this framework, some processes are limited by Gen. Harmonic Serialism allows only a single harmonically-improving step to be taken in each pass through the grammar. This restriction is enforced by Gen, which only generates candidates making a single change at each step.

2.2.1.1 Tonal steps in Gen

The following is a list of relevant steps which occur in this analysis:

(56) Tonal Steps in Gen

a. Create a new association line (violate *Assoc)

b. Delete an association line (violate *Dissoc)

c. Delete a high tone (violate Max-H)

d. Insert a high tone[5] (violate Dep-H)

Each of these steps violates a core faithfulness constraint, but may violate additional positional faithfulness constraints.

Two possible steps in Gen are the association and dissociation of high tones. *Association assigns a violation when a new association line is created, shown in (57):

(57) Create a new association line (violate *Assoc)

a. $tá_1ka$ → $tá_1ká_1$

b. H

taka → $tá_1ka$

The creation of a new association line can be the spreading of an existing associated high tone to an additional TBU, as in (a), or the association of a floating high tone, as in (b).

The dissociation of high tones involves the deletion of association lines, shown in (58):

[5] Tone insertion does not occur in this analysis, but would be a possible step.

(58) Delete an association line (violate *Dissoc)

a. $tá_1ká_1$ → $tá_1ka$

b.   H

$tá_1ka$ → taka

Similarly, this dissociation can be from a multiply linked high tone, as in (a), or from a singly linked high tone to a floating tone, as in (b).

In addition to association lines, high tones can also be deleted or inserted. An important distinction is that Max-H and Dep-H only affect floating tones. The deletion and insertion of floating high tones are shown in (59)-(60):

(59) Delete a high tone (violate Max-H)

H

a. taka → taka

(60) Insert a high tone (violate Dep-H)

    H

a. taka → taka

Only floating high tones can be deleted or inserted as a possible step in Gen. The deletion of associated high tones is a crucial aspect of this analysis, but it must occur in two independently motivated steps; first the dissociation of the high tone, followed by the deletion of the floating tone. Similarly, the deletion of a doubly linked high tone would require three separate steps in Gen: two steps of dissociation and one step of deletion.

2.2.1.2 Non-tonal steps in Gen

Non-tonal steps in Gen are less critical for this analysis because most of the steps relate to tonal interactions. For the most part these include a step which would violate a single faithfulness constraint. The most relevant non-tonal interactions are those relating to parsing and syllabification, which are discussed at the word level. In this analysis, parsing and syllabification are considered free steps, so they can occur with an additional faithfulness violation. Steps leading to resyllabification due to mora interaction are derived gradually, though this will be discussed in detail in the word level analysis.

2.2.2 The levels

There are a distinct set of processes which apply at each level. These processes must be ordered in their respective levels to account for the Kikerewe data. This section will give a summary of the processes that occur at each level, and why their ordering is able to fully account for the data.

(61) Processes at each level

Stem:	Deletion of consecutive high tones (MR)
Word:	Tone shift (TSH), Syllabic fusion (SF)
Phrase:	Lapse avoidance (LA), Tone spread (TSP)

2.2.2.1 MR applies at the stem level

An important process in Kikerewe is its application of Meeussen's Rule, which deletes consecutive high tones. As seen in earlier sections, some sequences of consecutive high tones are deleted, while others are not. Consider the following input→ output mappings:

(62) /abatá$_1$:-tú$_2$-gí$_3$-kú$_4$-hé$_5$:le:zye/ → [abatá$_1$:tú$_1$gikuhe:le:zye] 'they who didn't give it to us for you (remote)'

(63) /á$_1$-ku-chú$_2$mita/ → [akú$_1$chú$_2$mí$_2$ta] 'he who is stabbing'

In (62) the consecutive high tones in the underlying form are deleted, while in (63) new consecutive high tones are created in the surface form. The form with consecutive high tones in the surface form crucially relies on the ordering of MR before the creation of this HH sequence. As each of these processes requires a different constraint ranking, they must be separated into separate levels of the grammar.

With MR applying at the stem level, it will precede later processes that are necessary for the counterfeeding interaction. Namely, tone shift and syllabic fusion can be ordered at a later level where MR cannot delete new sequences of consecutive high tones. At the stem level, MR will be able to delete all the underlying cases of consecutive high tones.

2.2.2.2 Tone shift and syllabic fusion apply at the word level

With MR applying at the stem level, it is crucial that tone shift and syllabic fusion are ordered at a later level. Their application at the word level allows the grammar to capture the counterfeeding relationship with MR, with the intermediate forms shown here:

(64) /á$_1$-ku-chú$_2$mita/ → akú$_1$chú$_2$mita ‘he who is stabbing’

(65) /a-kí$_1$-a-ká$_2$laːnga/ → achá$_1$ːká$_2$laːnga ‘he is still frying’

With the processes of tone shift and syllabic fusion occurring at the word level, the grammar is able to capture the distinction between levels of the grammar in which consecutive high tones are deleted and in which they are maintained.

2.2.2.3 Lapse avoidance and tone spread occur at the phrase level

Lapse avoidance is a process which occurs between words in a phrase, so it is not particularly controversial that should be ordered at the phrase level. More significant is the interaction between lapse avoidance and tone spread at the phrase level. Recall the following data, which required different orderings of lapse avoidance and tone spread:

(66) /kubala kwa:ko/ → [kubalá$_1$ kwá$_1$:ko] ‘your (act of) counting’

(67) /ku:mbá$_1$la kwa:ko/ → [ku:mbá$_1$lá$_1$ kwa:ko] ‘your (act of) counting me’

In (66) lapse avoidance must apply first because the inserted tone then spreads, and in (67) tone spread must apply first to block lapse avoidance. As will be shown in the phrase level analysis, it is possible to derive both of these mappings with a single constraint ranking. Because this can capture all instances of tone spread, the most parsimonious analysis is to have both these processes co-occur at the phrase level. While a rule-based analysis would require tone spread to be ordered before and after lapse avoidance, the serial derivation and constraint-based interaction allow Harmonic Serialism to account for this data with a single constraint ranking.

2.2.3 Stem level

The first level of the grammar is the stem level. At this level, consecutive high tones are deleted, as in the form /abatá$_1$:-tú$_2$ –gí$_3$-kú$_4$-hé$_5$:le:zye/ ‘they who didn’t give it to us for you (remote)’ which results in the intermediate form /abatá$_1$:tugikuhe:le:zye/ as the output of the stem level.

The deletion of consecutive high tones is driven by the markedness constraint *HH (Odden 2000), which prohibits high toned autosegments associated with adjacent TBUs. This constraint has been used as a tone-specific version of the OCP constraint (Myers 1997), based on the Obligatory Contour Principle (Leben 1973; Goldsmith 1976).

The deletion of an associated high tone in Harmonic Serialism is a two step process: dissociation and deletion. The high tone must first dissociate from the TBU, violating *Dissociation, and then delete the floating high tone, violating Max-H. These steps require three crucial rankings: *HH>>*Float, *HH>>*Dissoc, and *Float>>Max-H. These rankings are shown in (68)-(69):

(68) Step 1: Dissociation

/$tá_1ká_2$/	*HH	*Float	*Dissoc	Max-H
a. $tá_1ká_2$	W_1	L	L	
b. H → $tá_1ka$		1	1	

(69) Step 2: Deletion

H /$tá_1ka$ /	*HH	*Float	*Dissoc	Max-H
a. H $tá_1ka$		W_1		L
b. → $tá_1ka$				1

The Markedness>>Faithfulness rankings allow the steps of dissociation and deletion to be harmonically improving. The ranking of two markedness constraints, *HH>>*Float, is crucial to allow the repair of *HH to be ordered before the repair of *Float.

With the current ranking it is possible to delete either the right or the left high tone. To allow only dissociation and deletion of the right tone, I propose a positional faithfulness constraint *Dissoc(left), which assigns a violation if the leftmost of two consecutive high tones is dissociated.

(70) *Dissoc(left): Do not dissociate a high tone in the output if it is left of another TBU with a high tone association.

*Dissoc(left) is a positional faithfulness constraint (Beckman 1997, 1998), which assigns violations with positions determined in the input. The use of positional faithfulness constraints in Harmonic Serialism has been motivated, solving pathological problems in Parallel OT (Jesney 2011).

The use of the *Dissoc(left) constraint allows the grammar to account for the directionality effect of MR. In the case of inputs with consecutive high tones, the left deletion candidate is harmonically bounded by the right deletion candidate, as shown in tableaux (71)-(72):

(71) Step 1: Dissociation of rightmost high tone

/tá$_1$ká$_2$/	*Dissoc(left)	*HH	*Float	*Dissoc	Max-H
a. tá$_1$ká$_2$		W$_1$	L	L	
b. H → tá$_1$ka			1	1	
c. H taká$_1$	W$_1$		1	1	

(72) Step 2: Deletion of floating high tone

H /$tá_1$ka /	*Dissoc(left)	*HH	*Float	*Dissoc	Max-H
a. H $tá_1$ka			W_1		L
b. → $tá_1$ka					1

The use of the *Dissoc(left) constraint selects candidate (b) with dissociation of the right high tone over candidate (c) with dissociation of the left high tone.

The *Dissoc(left) constraint is also crucially ranked above *HH. This ranking is needed to account for forms with more than two consecutive high tones, which would otherwise prefer to delete the medial high tones. This is illustrated in the following tableau:

(73) Step 1: No dissociation of medial high tone

/$pá_1tá_2ká_3$/	*Dissoc(left)	*HH	*Float	*Dissoc	Max-H
a. $pá_1tá_2ká_3$		W_2	L	L	
b. H → $pá_1tá_2$ka		1	1	1	
c. H $pá_1$ta$ká_2$	W_1	L	1	1	
d. H pa$tá_1ká_2$	W_1	1	1	1	

With this constraint ranking, the grammar selects candidate (b), with dissociation of the rightmost high tone. Candidate (c) gives the crucial ranking *Dissoc(left)>>*HH, which is necessary because deletion of medial high tones is able to reduce the violations of *HH by two, while deletion of initial or final high tones only reduces the number of *HH violations by one.

There have been several proposals in the literature to account for right to left high tone deletion in OT (Myers 1997; Cassimjee and Kisseberth 1998; Bickmore 1999; Odden 2000). However, they each run into problems which *Dissoc(left) avoids. The use of the *Dissoc(left) eliminates the problem noticed by (Odden 2000), that in the case of multiple consecutive high tones, a single violation of Max-H can repair multiple violations of *HH. Odden illustrated this problem with a Kikerewe form, which is shown in the following tableau, using a Parallel OT derivation:

(74) Medial deletion maximally reduces *HH violations

/abatá$_1$ː-tú$_2$-gí$_3$-kú$_4$-hé$_5$ːleːzye/	*HH	Max-H
a. abatá$_1$ː-tú$_2$-gí$_3$-kú$_4$-hé$_5$ːleːzye	W$_4$	L
b. ☹ abatá$_1$ː-tu-gi-ku-heːleːzye		$_4$
c. 💣 abatá$_1$ː-tu-gí$_2$-ku-hé$_3$ːleːzye		L$_2$

Deleting medial, rather than edge, high tones allows the grammar to reduce the violations of *HH with a minimal number of Max-H violations. Translated into a serial account, we have the same problem that requires directionality to derive the intended winner.

While the majority of Odden's analysis of Kikerewe is rule-based, he did account for some aspects of high tone deletion in a variant of OT. Odden (2000) proposed using a two-level constraint */H/H, which prohibits sequences of an underlying high toned TBU followed by a high tone. This constraint is problematic for two main reasons. First, it cannot account for all the data in Odden's analysis, due to over-deletion of shifted high tones, which he acknowledges. Second, this constraint introduces a two-level mechanism into the grammar, which may be unnecessary. There are independent reasons to be avoid including a two-level constraint, such as the lack of generalizations (McCarthy 1996). Furthermore, a two-level constraint is not compatible with Harmonic Serialism, because the input changes with each pass through the grammar. Therefore, the constraints would only have access to the input of the current pass, not the original input. Instead, I use the positional faithfulness constraint *Dissoc(left) and *HH.

The constraint ranking thus far ensures that high tones will dissociate in a right to left direction, as in (75):

(75) Step 1: Dissociation

/pá$_1$tá$_2$ka/	*Dissoc(left)	*HH	*Float	*Dissoc	Max-H
a. pá$_1$tá$_2$ka		W$_1$	L	L	
b. H → pá$_1$taka			1	1	
c. H patá$_1$ka	W$_1$		1	1	

As discussed, this ranking prevents deletion of non-rightmost high tones. The second step requires deletion of the floating high tone, which will require the use of *Assoc to prevent re-association of the floating high tone. The following tableau shows the second step of the derivation with the crucial ranking of *Assoc>>Max-H:

(76) Step 2: Deletion

H / pá$_1$taka /	*Dissoc(left)	*HH	*Float	*Assoc	*Dissoc	Max-H
a. H pá$_1$taka			W$_1$			L
b. → pá$_1$taka						1
c. pá$_1$taká$_2$				W$_1$		L

This ranking leaves deletion of floating high tones as the only possible repair.

The final ranking of the stem level constraints is shown in the following Hasse diagram:

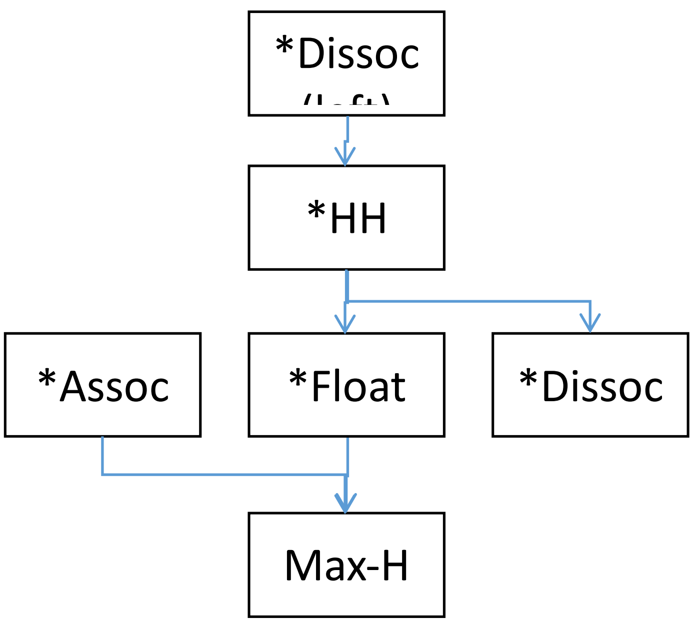

Figure 1: Stem level Hasse diagram

Using this ranking, the stem level form of /abatá$_1$:-tú$_2$-gí$_3$-kú$_4$-hé$_5$:le:zye/ to [abatá$_1$:tugikuhe:le:zye], as discussed by Odden (2000, 2008), can be derived. With five underlying high tones, this will take four steps of tone dissociation, four steps of floating tone deletion, and one step of convergence. This derivation is shown in (77)-(85):

(77) Step 1: Dissociation

/abatá$_1$:-tú$_2$-gí$_3$-kú$_4$-hé$_5$:le:zye/	*Dissoc(left)	*HH	*Float	*Assoc	*Dissoc	Max-H
a. abatá$_1$:tú$_2$gí$_3$kú$_4$hé$_5$:le:zye		W$_4$	L		L	
b. H → abatá$_1$:tú$_2$gí$_3$kú$_4$he:le:zye		3	1		1	
c. H abatá$_1$:-tú$_2$-gí$_3$-ku-hé$_4$:le:zye	W$_1$	L$_2$	1		1	

(78) Step 2: Dissociation

H /abatá$_1$:tú$_2$gí$_3$kú$_4$he:le:zye/	*Dissoc(left)	*HH	*Float	*Assoc	*Dissoc	Max-H
a. H abatá$_1$:tú$_2$gí$_3$kú$_4$he:le:zye		W$_3$	1		L	
b. H H → abatá$_1$:tú$_2$gí$_3$kuhe:le:zye		2	2		1	
c. abatá$_1$:tú$_2$gí$_3$kú$_4$he:le:zye		W$_3$	L		L	W$_1$

(79) Step 3: Dissociation

H H /abatá$_1$:tú$_2$gí$_3$kuhe:le:zye/	*Dissoc(left)	*HH	*Float	*Assoc	*Dissoc	Max-H
a. H H abatá$_1$:tú$_2$gí$_3$kuhe:le:zye		W$_2$	2		L	
b. H H H → abatá$_1$:tú$_2$gikuhe:le:zye		1	3		1	

(80) Step 4: Dissociation

H H H /abatá$_1$:tú$_2$gikuhe:le:zye/	*Dissoc(left)	*HH	*Float	*Assoc	*Dissoc	Max-H
a. H H H abatá$_1$:tú$_2$gikuhe:le:zye		W$_1$	3		L	
b. H H H H → abatá$_1$:tugikuhe:le:zye			4		1	

(81) Step 5: Deletion

H H H H /abatá$_1$:tugikuhe:le:zye/	*Dissoc(left)	*HH	*Float	*Assoc	*Dissoc	Max-H
a. H H H H abatá$_1$:tugikuhe:le:zye			W_4			L
b. H H H → abatá$_1$:tugikuhe:le:zye			3			1
c. H H H H H abata:tugikuhe:le:zye			W_5		W_1	L
d. H H H abatá$_1$:tugí$_2$kuhe:le:zye			3	W_1		L

(82) Step 6: Deletion

H H H /abatá$_1$:tugikuhe:le:zye/	*Dissoc(left)	*HH	*Float	*Assoc	*Dissoc	Max-H
a. H H H abatá$_1$:tugikuhe:le:zye			W_3			L
b. H H → abatá$_1$:tugikuhe:le:zye			2			1

(83) Step 7: Deletion

H H /abatá$_1$:tugikuhe:le:zye/	*Dissoc(left)	*HH	*Float	*Assoc	*Dissoc	Max-H
a. H H abatá$_1$:tugikuhe:le:zye			W$_2$			L
b. H → abatá$_1$:tugikuhe:le:zye			1			1

(84) Step 8: Deletion

H /abatá$_1$:tugikuhe:le:zye/	*Dissoc(left)	*HH	*Float	*Assoc	*Dissoc	Max-H
a. H abatá$_1$:tugikuhe:le:zye			W$_1$			L
b. → abatá$_1$:tugikuhe:le:zye						1

(85) Step 9: Converge

/abatá$_1$:tugikuhe:le:zye/	*Dissoc(left)	*HH	*Float	*Assoc	*Dissoc	Max-H
a. → abatá$_1$:tugikuhe:le:zye						
b. H abata:tugikuhe:le:zye					1	

In this derivation, the first four steps are dissociation of consecutive high tones. Due to the ranking of *HH>>*Float, all violations of *HH are repaired via dissociation before the floating high tones are deleted. After four of the five high tones have been dissociated, the remaining high tones are gradually deleted, one in each pass through the grammar. The grammar converges when only a single associated high tone remains in tableau (85).

At the stem level, the grammar deletes the sequences of consecutive high tones from right to left. The output from the stem level would then be passed on to the word level.

2.2.4 Word level

The second stratum in the grammar is the word level. The processes occurring at this level create sequences of consecutive high tones. Unlike in the stem level, these derived consecutive high tones are not deleted at the word level.

2.2.4.1 Onsetless tone shift

One process that creates consecutive high tones is the onsetless tone shift. In Kikerewe, when a high tone is associated with a word-initial onsetless syllable, it shifts one syllable to the right because onsetless syllables cannot bear tone (Odden 1995). Recall the following forms:

(86) a. /bá$_1$-ku-baziːla/ → bá$_1$kubaziːla → [bá$_1$kú$_1$baziːla] 'they who are sewing'

b. /á$_1$-ku-baziːla/ → akú$_1$baziːla → [akú$_1$bá$_1$ziːla] 'he who is sewing'

In (a), the tone remains on its original TBU and then spreads to the right via tone spread. In (b), the tone associated with an onsetless syllable shifts one syllable to the right and then undergoes tone spread.

The process of onsetless tone shift also interacts with neighboring high tones. This process can create sequences of consecutive high tones, which is seen in the following form:

(87) /$\acute{a}_1$-ku-ch$\acute{u}_2$mita/ → ak$\acute{u}_1$ch$\acute{u}_2$mita [6]

In this form, the high tone shifts from the initial onsetless syllable to the second syllable, causing it to be adjacent to the high tone on the third syllable. While consecutive high tones at the stem level were deleted, they are not deleted at the word level. The interaction here is underapplication opacity, with onsetless tone shift creating consecutive high tones that are not deleted.

While onsetless syllables have special behavior in Kikerewe, they are permitted word-initially. The Onset constraint, which assigns a violation to syllables without an onset, is not undominated. It is ranked with respect to Max-V, which assigns a violation for deleting a vowel, and Dep-C, which assigns a violation for inserting a consonant. There is a crucial ranking of Max-V, Dep-C>>Onset to eliminate candidates which satisfy the onset requirement. This is shown in tableau (88):

(88) Onsetless syllables permitted word-initially

/$\acute{a}_1$-ku-ch$\acute{u}_2$mita/	Max-V	Dep-C	Onset
a. → $\acute{a}_1$kuch$\acute{u}_2$mita			1
b. kuch$\acute{u}_1$mita	W_1		L
c. t$\acute{a}_1$kuch$\acute{u}_2$mita		W_1	L

[6] This form will become [ak$\acute{u}_1$ch$\acute{u}_2$m$\acute{i}_2$ta] at the phrase level.

The high ranking of the faithfulness constraints above onset results in the selection of the faithful form as the optimal candidate, without deletion of the initial vowel, as in (b), or insertion of an initial consonant, as in (c).

The special behavior of onsetless syllables in prosodic processes has been well-documented. The effects seen in stress assignment, reduplication, and tone association have been attributed to prosodic misalignment (Downing 1998, 1999). In addition to Downing's account of IsiXhosa, is that prosodic misalignment eliminates the violation of Onset, the effects of which are seen in the special behavior of onsetless syllables in reduplication. Downing also uses this phenomenon to explain onsetless syllables in Kikerewe. The initial onsetless syllable is unparsed to reduce a violation of onset, and unparsed syllables are dispreferred as TBUs.

I follow Downing (1999) in relying on the constraint PWord≈MWord (McCarthy and Prince 1993a) to account for prosodic misalignment, defined as follows:

(89) PWord≈MWord: PWord is coextensive with MWord.

For misalignment to occur, Onset must be ranked above P≈M. The parsing of /á-ku-chúmita/ is shown in tableau (90):

(90) Prosodic misalignment

/á_1-ku-chú_2mita/	Onset	P≈M
a. [{$\text{á}_1$kuchú$_2$mita }]	W_1	L
b. → [á_1{kuchú$_2$mita }]		1

In this tableau, candidate (b) with prosodic misalignment is preferred over candidate (a), which has a violation of Onset. In Harmonic Serialism, forms are derived gradually, with a single change at each step. I will assume for simplicity that parsing is free for the purposes of gradual changes, though this is not crucial to the analysis. Parsing does not count as a step in the derivation, but the optimal parse is selected at each step based on the current input and constraint ranking. In the Kikerewe data, words beginning with onsetless syllables will consistently remain parsed as the winner in (90), with prosodic misalignment of the onsetless syllable.

With the onsetless syllable unparsed, as in (90), I now account for the tone shift from the onsetless syllable. I will use the constraint Parse-H (Myers 1994), which requires that tones are parsed. This will assign a violation to a tone which is not associated with a parsed TBU. This will need to be crucially ranked above *Associate (Yip 2002), a constraint that assigns a violation for each new association line. This ranking will allow for spreading of the unparsed high tone.

(91) Tone spread from unparsed TBU

/á$_1$-ku-chú$_2$mita/	Onset	P≈M	Parse-H	*Assoc
a. [á$_1${kuchú$_2$mita}]		1	W$_1$	L
b. → [á$_1${kú$_1$chú$_2$mita}]		1		1
c. [{á$_1$kuchú$_2$mita}]	W$_1$	L		

This ranking selects candidate (b), which has spreading to remove the violation of Parse-H, as the optimal candidate.

At the word level, the creation of consecutive high tones is allowed. Tone spread from an unparsed TBU can incur a violation of *HH, as shown in (92):

(92) Tone spread can create consecutive high tones

/á$_1$-ku-chú$_2$mita/	Onset	P≈M	Parse-H	*Assoc	*HH
a. [á$_1${kuchú$_2$mita}]		1	W$_1$	L	L
b. → [á$_1${kú$_1$chú$_2$mita}]		1		1	1
c. [{á$_1$kuchú$_2$mita}]	W$_1$	L			L

The ranking of *HH is sufficiently low such that it does not block the process of tone shift.

In the second step of the analysis, the initial tone which has just spread delinks from the unparsed syllable. This requires a constraint *Unparsed-H, which assigns a violation to a tone associated with an unparsed syllable.

(93) *Unparsed-H: Assign a violation to a high tone associated with an unparsed syllable.

*Unparsed-H must be ranked above *Dissociate (Yip 2002), which assigns a violation for removing an association line, in order to allow dissociation of the high tone. This is illustrated in tableau (94):

(94) Dissociation from unparsed syllable

/[á$_1${kú$_1$chú$_2$mita}]/	*Unparsed-H	*Dissoc
a. [á$_1${kú$_1$chú$_2$mita}]	W$_1$	L
b. → [a{kú$_1$chú$_2$mita}]		1

With this ranking, the grammar selects candidate (b), which dissociates from the unparsed syllable, resulting in a singly linked tone on the first parsed syllable. These constraints need to be ordered with respect to the constraints needed for step 1.

The constraints in step 2, *Unparsed-H and *Dissociate, must be ordered below Parse-H in order for step 2 to be ordered after step 1. If *Unparsed-H were ranked above Parse-H, it would be ordered first. The correct derivation of the first two steps, with the ranking Parse-H>>*Unparsed-H is shown below:

(95) Step 1

/á$_1$-ku-chú$_2$mita/	Onset	P≈M	Parse-H	*Unparsed-H	*Dissoc	*Assoc	*HH
a. [á$_1${kuchú$_2$mita}]		1	W$_1$	1		L	L
b. → [á$_1${kú$_1$chú$_2$mita}]		1		1		1	1

(96) Step 2

/[á$_1${kú$_1$chú$_2$mita}]/	Onset	P≈M	Parse-H	*Unparsed-H	*Dissoc	*Assoc	*HH
a. [á$_1${kú$_1$chú$_2$mita}]		1		W$_1$	L		L
b. → [a{kú$_1$chú$_2$mita}]		1			1		1

The winner in step 1 associates to the following syllable to satisfy Parse-H, and the winner in step 2 dissociates from the initial unparsed syllable to satisfy *Unparsed-H.

With no remaining harmonically improving steps, this winner converges on the next pass through the grammar:

(97) Step 3: Converge

/[a{kú$_1$chú$_2$mita}]/	Onset	P≈M	Max-H	Parse-H	*Unparsed-H	*Dissoc	*Assoc	*HH
a. →[a{kú$_1$chú$_2$mita}]		1						1
b. H [a{kú$_1$chumita}]		1				W$_1$		L

The winning candidate has a remaining violation of the markedness constraint, *HH. This violation cannot be repaired, with the crucial ranking of *Dissoc>>*HH. The fully faithful form is selected as the winner.

2.2.4.2 Syllabic fusion

Another process undergone at the word level is syllabic fusion, which occurs to repair word internal onsetless syllables. Syllabic fusion encompasses a set of processes, vowel gliding and compensatory lengthening. This is shown in the following data from Odden (1995):

(98) a. ebi-to:ke 'bananas' /ebi-ála/ → ebj-á:la 'fingers'
b. omu-tíma 'heart' /omu-aga/ → omw-a:ga 'compulsion'
c. olu-bíbo 'fish trap' /olu-íle/ → olw-í:le 'sky'

For each of these forms containing two adjacent vowels underlyingly, in the surface form, the first vowel is a glide and the second vowel is lengthened, resulting in a single syllable.

In the Harmonic Serialism level, the first step of vowel gliding occurs to remove the violation of Onset. As in the previous section, there is the crucial ranking of Onset>>

P≈M. Additionally, there is a crucial ranking of Onset and some faithfulness constraint governing vowels. This can be illustrated with a stand-in constraint Id-V, which assigns a violation when the identity of vowel is changed, giving the ranking of Onset>>Id-V. This ranking is shown in the following tableau:

(99) Vowel gliding repairs word-medial onsetless syllables

/omu-aga/	Onset	P≈M	Id-V
a. [o{mu}a{ga}]		W_2	L
b. → [o{mwaga}]		1	1
c. [o{muaga}]	W_1	1	L

The winner, candidate (b), repairs the violation of Onset by gliding the medial onsetless vowel, incurring a violation of Id-V. While this simplified version shows the process of vowel gliding in a single step, the full analysis will require multiple harmonically improving steps.

The process of vowel gliding is accompanied by compensatory lengthening, which benefits from a moraic analysis to simultaneously account for these two processes. This can be done gradually with steps of mora delinking and relinking, shown in (100):

(100) Gradual mora association

μμ μμ μμ
| | | \/
omuaga → omwaga → omwa:ga

The first step of mora delinking results in the formation of a glide from the vowel. In the next step the remaining vowel links to the floating mora, resulting in compensatory lengthening.

Following an OT-CC analysis of compensatory lengthening (Shaw 2009), I will use the faithfulness constraints DepLink-μ[7] and MaxLink-μ (Morén 2000):

(101) Moraic constraints

DepLink(Mora,Segment): for moras and segments in correspondence, mora-to-segment links in the output correspond to mora-to-segment links in the input.

MaxLink(Mora,Segment): for moras and segments in correspondence, mora-to-segment links in the input correspond to mora-to-segment links in the output.

Ranking these constraints, with an additional markedness constraint against floating moras, the grammar undergoes syllabic fusion in two steps:

(102) Step 1: Vowel gliding

/omu$_\mu$-aga/	Onset	P≈M	MaxLink	*Float- μ	DepLink
a. [o{mu$_\mu$}a{ga}]		W_2	L	L	
b. μ → [o{mwaga}]		1	1	1	
c. [o{mu$_\mu$aga}]	W_1	1	L	L	

[7] This constraint is the same as No-Spread(τ, ς) (McCarthy 2000b).

(103) Step 2: Compensatory lengthening

µ /omwaga/	Onset	P≈M	MaxLink	*Float- µ	DepLink
a. µ [o{mwaga}]		1		W_1	L
b. µ | → [o{mwa:ga}]		1			1

In the first step, the crucial ranking of Onset>>MaxLink allows dissociation of the mora, resulting in vowel gliding. In the second step, the ranking of *Float- µ>>DepLink associated the floating mora with the vowel, resulting in compensatory lengthening.

Thus far I have established the ranking for syllabic fusion, summarized in the following Hasse diagram:

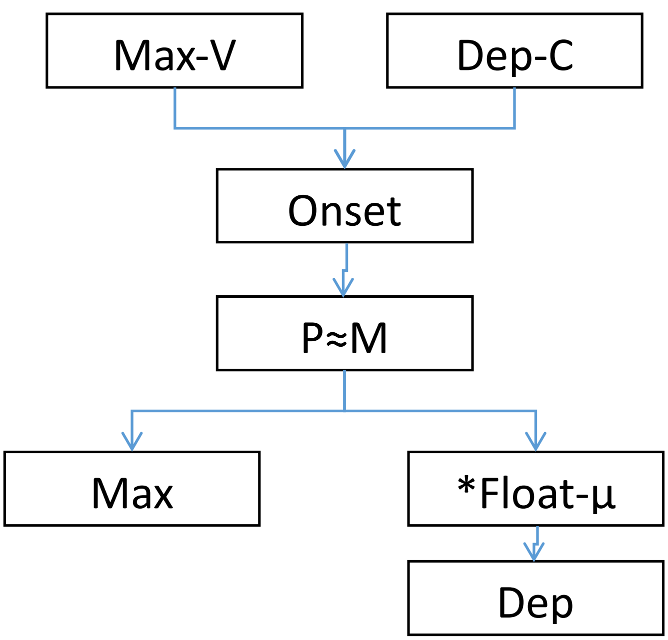

Figure 2: Partial Hasse diagram

Given these rankings for syllabic fusion, I will now consider a case where syllabic fusion interacts with tonal processes. Recall that the process of syllabic fusion can create consecutive high tones, as seen in the following form:

(104) /a-kí$_1$-a-ká$_2$la:nga/ → [achá$_1$:ká$_2$lá$_2$:nga][8]

In this example, the process of syllabic fusion causes the two high tones in the underlying form to become adjacent, resulting in a violation of *HH. As in the case of tone shift, these consecutive high tones are not deleted, but instead result in surface opacity. The relevant intermediate forms at the word level are shown in (105):

(105) /a-kí$_1$-a-ká$_2$la:nga/ → akjá$_1$ká$_2$la:nga → achá$_1$:ká$_2$la:nga

The initial step is glide formation, to repair the medial onsetless syllable. While the high tone appears to have shifted, as the syllable is the TBU in Kikerewe, the fusion of the two vowels into a single syllable gives us this step for free. The next intermediate form shows compensatory lengthening, which will be derived in two steps in the grammar, via mora linking and delinking. The final form shows the process of tone fusion, and an additional step of palatalization, though these processes are not ordered with respect to each other. The final step of tone spread, as seen in the surface form [achá$_1$:ká$_2$lá$_2$:nga], will not be accounted for in this section, as it is a phrase level process. Most of these processes can be accounted for with the constraint rankings already proposed, though a few additional rankings will be needed.

The first new ranking needed will include the constraint *HH. In the first step of the derivation, vowel gliding occurs to repair the word-medial onsetless syllable. This process results in resyllabification of the onsetless vowel into the preceding syllable, in

---

[8] This surface form is shown with tone spread.

turn creating a sequence of consecutive high tones that violate *HH. This violation gives a crucial ranking of Onset, P≈M>>*HH, as follows:

(106) Step 1: Vowel gliding can create *HH violation

/a.kí$_1$.a.ká$_2$.la:.nga/	Onset	P≈M	MaxLink	*Float- μ	DepLink	*HH
a. [a{kí$_1$}a{ká$_2$la:nga}]		W$_2$	L	L		L
b. μ → [a{kjá$_1$.ká$_2$la:nga}]		1	1	1		1
c. [a{kí$_1$.a.ká$_2$la:nga}]	W$_1$	1				L

The winning candidate removes the violation of Onset and P≈M by delinking the onsetless syllable from the mora, incurring a violation of the lower ranked markedness constraint *HH. As *HH is not crucially ranked with respect to the moraic constraints, we will rank it below them for the ordering of steps, but this is not crucial.

In the next pass through the grammar, compensatory lengthening occurs due to linking of the floating mora:

(107) Step 2: Compensatory lengthening

μ /akjá$_1$.ká$_2$la:nga/	Onset	P≈M	MaxLink	*Float- μ	DepLink	*HH
a. μ [a{kjá$_1$.ká$_2$la:nga}]				W$_1$	L	1
b. μ | → [a{kjá:$_1$.ká$_2$la:nga}]					1	1

Candidate (b) removes the violation of *Float- μ and is selected as the winner.

In the final pass through the grammar, the derivation will converge:

(108) Step 3: Converge

/achá$_1$:ká$_2$la:nga/	Onset	P≈M	MaxLink	*Float-μ	DepLink	*Dissoc	*HH
a. → [a{chá$_1$:ká$_2$la:nga}]							1
b. H [a{chá$_1$:kala:nga}]						W$_1$	L

The final markedness violation of *HH cannot be repaired, which gives a crucial ranking of *Dissoc>>*HH. With no remaining harmonically improving steps, the derivation converges on [achá$_1$:ká$_2$la:nga]. Note that there is an additional step of palatalization seen in this form. This can be accounted for with a simple ranking of a markedness constraint, *kj, over a faithfulness constraint, Id-C, but this has been omitted for brevity, as it is not crucial to the tonal representations.

2.2.4.3 High tone morpheme

Recall the process of lapse avoidance, which inserts a high tone before a toneless modifier in a noun-modifier phrase (Odden 2008):

(109) a. oluguhyo 'broken pot'

b. luːkizaːno 'green'

c. luzímá 'good'

d. oluguhyó lúːkizaːno

e. oluguhyo luzímá

The modifier in (b) is toneless, and the corresponding phrase in (d) has high tone insertion. The modifier in (c) contains a high tone, and the corresponding phrase in (e) does not undergo tone insertion. Based on this data, the analysis will need to account for the distribution of the high tone in noun-modifier phrases.

Thus far, the details of how the high tone is inserted have not been laid out. While Odden (2000, 2008) accounts for the distribution of lapse avoidance, he does not propose an analysis for the insertion of this high tone. In this section, I will begin to provide a constraint-based account of lapse avoidance.

I will consider this high tone to be the result of a linking morpheme consisting of a floating high tone. This morpheme affixes to the modifier at the word level, but is deleted in the presence of another high tone. To accomplish this, I will use a constraint to penalize the toneless morpheme in the presence of another high tone:

(110) OCP-H: Assign a violation for multiple high tones in the same domain

This constraint is crucially ranked above Max-H, which allows deletion of the floating tone when it co-occurs with a high-toned modifier, as shown in the following tableau:

(111) Floating tone deleted with toned modifier

/H luzimá$_1$/	OCP-H	Max-H
a. H luzimá$_1$	W$_1$	L
b. → luzimá$_1$		1

The winner, candidate (b), has deletion of the floating high tone due to repair the violation of OCP-H.

Deletion of the floating high tone does not occur if the modifier is toneless. The constraint prohibiting floating tones must therefore be ranked below Max-H, shown in (112):

(112) Floating tone maintained with toneless modifier

/H lu:kiza:no/	OCP-H	Max-H	*Float
a. → H lu:kiza:no			1
b. lu:kiza:no		W$_1$	L

In this tableau, it is not harmonically improving for the high tone to be deleted. The input has no initial violations of OCP-H, and repair of the *Float constraint incurs a violation of the higher ranked Max-H.

As seen in earlier examples at the word level, it is possible to have words containing two or more high tones which are not deleted. It would be undesirable for these tones to be deleted due to OCP-H. However, due to the gradual nature of associated tone deletion, this is not predicted. Consider the following tableau, with the hypothetical input containing two high tones which will not be deleted at the word level:

(113) OCP-H cannot trigger deletion of associated high tones

/pá$_1$taká$_2$/	OCP-H	Max-H	*Float	*Dissoc
a. → pá$_1$taká$_2$	1			
b. H pá$_1$taka	1		W$_1$	W$_1$

While this form has a violation of OCP-H, it cannot be repaired in a single step. Deletion of an associated high tone requires an initial step of tone dissociation, followed by a step of tone deletion. The loser is harmonically bounded by the winner because the initial step of tone dissociation does not remove the violation of OCP-H, though it does incur additional violations of *Float and *Dissoc. The loser is not harmonically improving and none of the high tones are deleted. While this ranking can account for the distribution of the morpheme with respect to toneless modifiers, the association of the floating high tone will occur at the phrase level.

With these rankings, the Hasse diagram for the word level is summarized in the following figure:

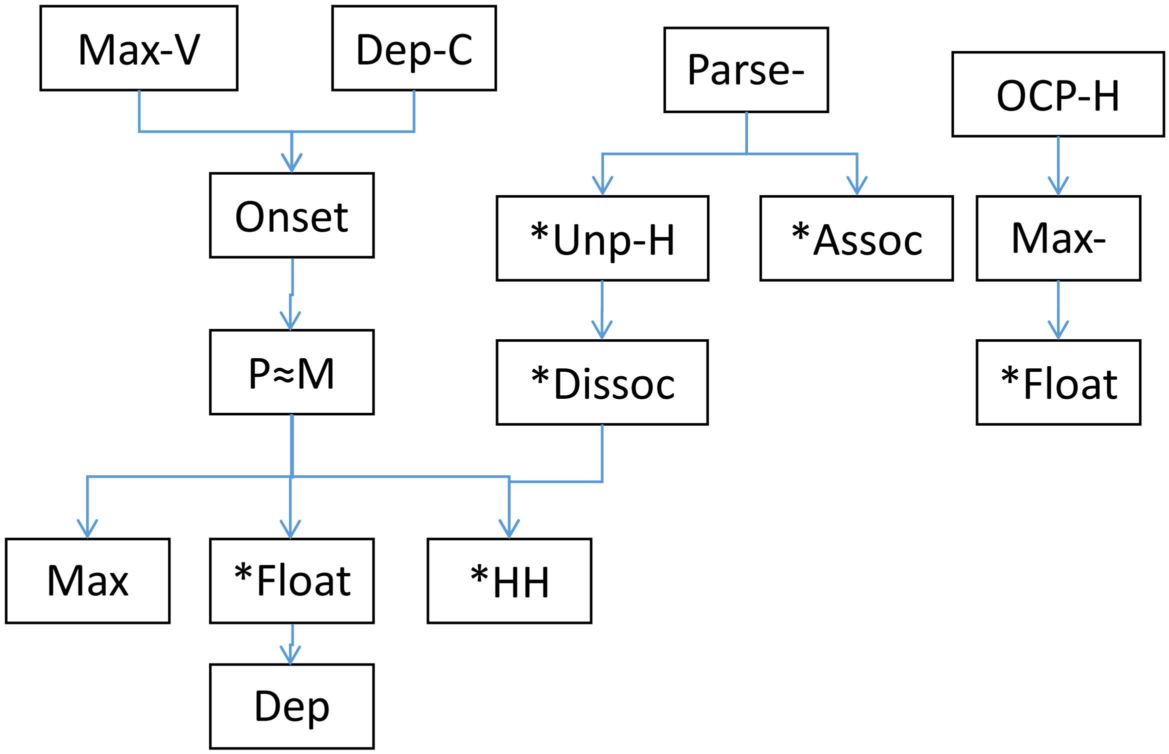

Figure 3: Word level rankings

2.2.5 Phrase level

The phrase level is the final level of the grammar. Important processes that occur at this level are tone spread and lapse avoidance.

2.2.5.1 Tone spread

Tone spread involves the association of a singly-linked high tone to an additional TBU due to a violation of the markedness constraint Tone Minimality (Bickmore 1996; Odden 2000), which prefers doubly-linked high tones. While the original version only penalized output forms, I have slightly revised the definition to be sensitive to input forms as well. The new definition is shown in (114):

(114) Tone Minimality (revised): Assign a violation to each high tone in the input associated with fewer than two TBUs.

This constraint makes reference to high tones in both the input and the output. This means that violations of Tone Minimality are removed by association to a new TBU, but not by deletion. While the original version differed somewhat, it was also used in Parallel OT, rather than Serial OT. This may require the adjustment of some constraints.

The process of tone spread requires the crucial ranking of ToneMin>>*Assoc, as shown in the following hypothetical case:

(115) Satisfaction of ToneMin via spreading

/pá_1taka/	ToneMin	*Assoc
a. pá_1taka	W_1	L
b. → $\text{pá}_1\text{tá}_1$ka		1

With this ranking, the candidate with a doubly-linked high tone is more harmonic than the faithful candidate with a singly-linked high tone.

The process of tone spread affects singly-linked tones which were present in the output of word level forms. Recall the form *akú₁chú₂mita*, generated at the word level via tone shift. At the phrase level, this form undergoes tone spread, shown in (116):

(116) Tone spread

/$\text{akú}_1\text{chú}_2$mita/	ToneMin	*Assoc
a. $\text{akú}_1\text{chú}_2$mita	W_2	L
b. → $\text{akú}_1\text{chú}_2\text{mí}_2$ta	1	1

This ranking selects the intended winner, which is the surface form [akú$_1$chú$_2$mí$_2$ta]. While this does answer allow the correct harmonically improving step for the second high tone, we must also explain the directionality of the spread and the absence of spreading for the first high tone.

In the case of *akú$_1$chú$_2$mita*, rightward spreading of the second high tone is necessary because the adjacent high tone would block any potential leftward spreading. However, additionally we do not see leftward spreading in the first high tone. While this could potentially be due to a restriction on spreading to onsetless syllables, other forms show us that this must be a restriction on leftward spreading.

First, I will establish that rightward spreading is preferred to leftward spreading. When both leftward and rightward spreading are possible, rightward spreading is the selected operation. Consider the following data, which shows rightward spreading in a word-medial high tone (Odden 2000):

(117) Rightward spreading preferred to leftward spreading

a. ku-bala ‘to count’

b. ku-tú$_1$-bá$_1$la ‘to count us’

The form in (a) shows that the verbal stem *bala* is toneless, but the form in (b) has a high tone on personal pronoun morpheme, *tú*, which spreads rightward to the verbal stem. Unlike the form *akú$_1$chú$_2$mí$_2$ta*, this form has no onsetless syllable to restrict leftward spreading, but spreading is still rightward.

The preference for rightward over leftward spreading can be attributed to a ranking of positional faithfulness constraints, *Assoc-L>>*Assoc-R. This ranking will prefer rightward spreading to leftward spreading, if both are possible, as illustrated in the following tableau:

(118) Rightward spreading preferred over leftward spreading

/kutú$_1$bala/	Tone Min	*Assoc-L	*Assoc-R	*Assoc
a. kutú$_1$bala	W$_1$		L	L
b. → kutú$_1$bá$_1$la			1	1
c. kú$_1$tú$_1$bala		W$_1$	L	1

In this tableau, candidate (b) with rightward tone spread is preferred over leftward spreading as the repair for ToneMin.

Next, we will consider whether leftward spreading is possible when rightward spreading is blocked. This will give us the relative ranking of *Assoc-L and ToneMin. There is a specific environment in which rightward spreading is not possible, but leftward spreading will still not occur to repair the violation of ToneMin, leaving a singly-linked high tone as the output of the phrase level. Evidence for the prohibition of leftward spreading comes from forms with underlying penultimate high tones. Consider the following data (Odden 2000), which does not undergo tone spread:

(119) Non-finality and no leftward spreading

a. ku-sya ‘to grind’

b. ku-tú$_1$-sya ‘to grind us’

Like the data in (117), this minimal pair shows the interaction of a high-toned pronoun with a toneless verbal stem. While the previous example had a disyllabic verbal stem, here there is a monosyllabic stem, which does not undergo tone spread. The blocking of tone spread is due to a Non-finality constraint, which prohibits high-toned final TBUs.

While this constraint blocks rightward spreading to a final TBU, it does not trigger deletion of pre-existing final high tones, as in the verbal stem *lyá₁* 'eat.'[9]

To prevent rightward tone spread, I will use the NonFinality constraint (Odden 2008), which assigns a violation when a high tone is associated with a final TBU.

(120) NonFinality: Assign a violation when a high tone is associated with the final TBU

The blocking of tone spread to a final syllable requires the ranking NonFinality>>ToneMin, which is shown in tableau (121):

(121) Non-finality blocks rightward tone spread

/ kutú$_1$sya /	Non-Fin	*Dissoc	Tone Min	*Assoc-L	*Assoc-R	*Assoc
a. →kutú$_1$sya			1			
b. kutú$_1$syá$_1$	W$_1$		L		W$_1$	W$_1$
C. H kutusya		W$_1$	L			

Candidate (b) with rightward tone spread is not harmonically improving, and is eliminated by the high-ranked NonFinality constraint. Candidate (c) with tone dissociation is eliminated, giving the crucial ranking of *Dissoc>>ToneMin, which does not allow dissociation as a repair for singly-linked high tones.

With the prohibition of rightward spreading to final syllables, the ranking of *Assoc-L can be further established. The ranking of *Assoc-L>>*Assoc-R is not sufficient because leftward spread does not occur when rightward spread is independently

[9] This verbal stem can undergo leftward spreading at the phrase level, as in the form *kú₁-lyá₁* 'to eat'

blocked. There is an additional crucial ranking of *Assoc-L>>ToneMin, which prevents the repair of singly-linked tones via leftward spreading at the Word level. This is shown in tableau (122):

(122) No leftward spreading

/ kutú$_1$sya /	NonFin	*Dissoc	*Assoc-L	ToneMin	*Assoc-R	*Assoc
a. →kutú$_1$sya				1		
b. kutú$_1$syá$_1$	W$_1$			L	W$_1$	W$_1$
c. H kutusya		W$_1$				
d. kú$_1$tú$_1$sya			W$_1$	L		W$_1$

In this tableau, candidate (d) is eliminated because the step of leftward spreading is not harmonically improving, due to the high ranking of *Assoc-L. Candidate (a) without spreading is the winner.

Having established the general trend of rightward spreading, I will briefly outline the only case in which leftward spreading is permitted. As shown in the previous example, Kikerewe has a generalization, governed by the NonFinality constraint, in which high tones are dispreferred on final syllables. While high tones cannot spread to final syllables, those which appear underlying are not deleted. This situation most frequently occurs in the case of high-toned monosyllabic verbal stems, such as the following:

(123) Monosyllabic high-toned verbal stems

a. lyá 'eat'

b. há 'give'

In addition to not being deleted, the high tones in these verbal stems can undergo leftward spreading if they appear phrase finally and are preceded by a toneless syllable. The distinction between toned and toneless verbal stems can be seen in the following data, where the high toned morpheme undergoes leftward spreading:

(124) Toneless vs. toned monosyllabic verbal stems

a. /ku-sya/ → ku-sya 'to grind'

b. /ku-lyá$_1$/ → kú$_1$-lyá$_1$ 'to eat'

While leftward spreading is possible when the high tone appears phrase finally, rightward spreading is still preferred when the high tone is not final:

(125) Phrase final spreading

a. kú$_1$–há$_1$ 'to give'

b. ku-há$_1$ Bú$_1$lemo 'to give Bulemo'

The form in (a) has leftward spreading from the high tone on the verbal stem *há$_1$* to the underlyingly toneless morpheme *ku*. In this form, leftward spread occurs because the high tone is final. The form in (b) has the same monosyllabic high-toned verbal stem, but it undergoes rightward spreading. This form is also significant because it shows that the domain of tone spread is the phrase, not the word. As seen in (125), the same word /ku-há$_1$/ can have different surface tones depending on the phrasal context.

Based on this data, we will need to account for the possibility of leftward tone spread when a high tone appears finally, but not in other cases. This will also need to exclude leftward spreading even in cases where rightward spreading is blocked, as shown in the form *kutú₁sya*, discussed in (122). With the current ranking, shown in (126), the grammar will not select the intended winner:

(126) Current constraint set insufficient

/ ku-$há_1$/	NonFin	*Dissoc	*Assoc-L	ToneMin	*Assoc-R	*Assoc
a. ku-$há_1$	1			W_1		
b. $kú_1$-$há_1$	1		1			1
c. H ku-ha	L	W_1		W_1		

The current set of constraints is not sufficient to account for this set of data. To account for this, I will use a positional markedness constraint which penalizes final high tones that are singly linked, similar in form to a conjoined constraint of ToneMinimality and NonFinality, FinalMinimality.

(127) FinalMinimality: Assign a violation to a singly linked high tone on the final TBU of the input.

This constraint assigns a violation to a high tone which is only linked to a final TBU. Using this constraint, we can get the final ranking to account for tone spread, shown in (128):

(128) Final tone spread ranking

/ ku-$há_1$/	*Dissoc	FinMin	NonFin	*Assoc-L	ToneMin	*Assoc-R	*Assoc
a. ku-$há_1$		W_1	1	L	W_1		
b. → $kú_1$-$há_1$			1	1			1
c. H ku-ha	W_1		L	L	W_1		

FinMin is crucially ranked above *Assoc-L, as shown by candidate (a). Also, *Dissoc is ranked above NonFin, as shown by candidate (c). With this final ranking, we can account for all directional tone spreading at the phrase level.

With the rankings of all the constraints determined, I will now return to the discussion of *ak*$ú_1$*ch*$ú_2$*m*$í_2$*ta* and its derivation at the phrase level. In the first step, the second high tone spreads rightward:

(129) Step 1: Spread right

/akú$_1$chú$_2$mita/	*Dissoc	FinMin	NonFin	*Assoc-L	ToneMin	*Assoc-R	*Assoc
a. akú$_1$chú$_2$mita					W$_2$		L
b. → akú$_1$chú$_2$mí$_2$ta					1		1
c. á$_1$kú$_1$chú$_2$mita				W$_1$	1		1
d. H akú$_1$chumita	W$_1$				W$_2$		L

In this step, the winner has rightward spreading of a singly-linked high tone. Candidate (c), which has leftward spreading is blocked by the high ranked *Assoc-L. Candidate (d) is eliminated by violation of the high ranked *Dissoc.

In the next pass through the grammar, this form will converge:

(130) Step 2: Converge

/ akú$_1$chú$_2$mí$_2$ta /	*Dissoc	FinMin	NonFin	*Assoc-L	ToneMin	*Assoc-R	*Assoc
a. → akú$_1$chú$_2$mí$_2$ta					1		
b. á$_1$kú$_1$chú$_2$mí$_2$ta				W$_1$	L		1

In this step, the fully faithful candidate is the winner. Candidate (b) fatally violates *Assoc-L, leaving candidate (a) with an unresolved ToneMin violation. This gives us the correct surface form [akú$_1$chú$_2$mí$_2$ta].

Thus far, the ranking arguments for the phrase level are shown in the following Hasse diagram:

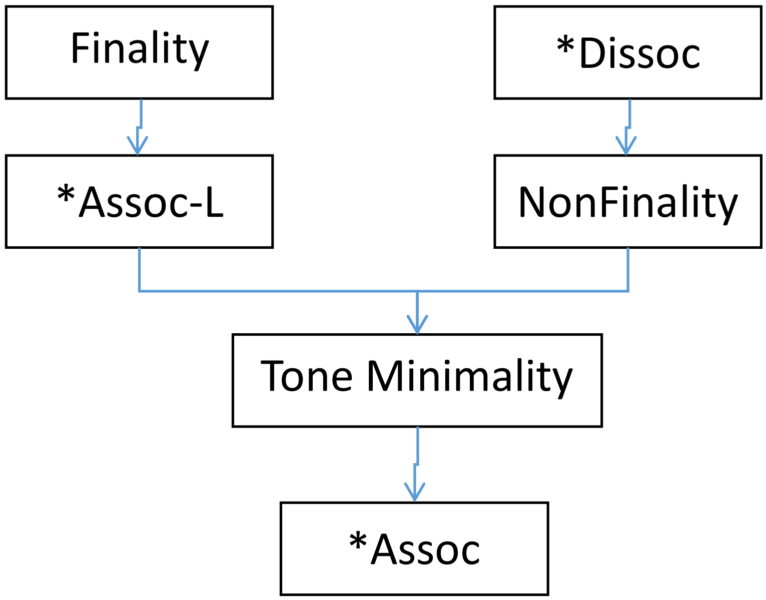

Figure 4: Partial phrase level ranking

2.2.5.2 Lapse avoidance

The process of lapse avoidance also occurs at the phrase level. Odden identified this process of tone insertion on the final syllable of a noun when followed by a toneless modifier. This high tone then undergoes the regular process of spreading, as seen in (131):

(131) Lapse avoidance

a. oluguhyo ‘broken pot’

b. lu:kiza:no ‘green (Cl. 11)’

c. oluguhyó$_1$ lú$_1$:kiza:no ‘green broken pot’

In this analysis, I attribute the process of lapse avoidance to a high tone morpheme, which links a noun and a modifier in a phrase. In the underlying representation, this morpheme is a floating high tone, which associates to a TBU at the phrase level. This high tone morpheme only surfaces with toneless modifiers. Recall that when co-occurring with toned modifiers, the floating high tone deletes at the word level due to an OCP-H constraint.

This morphological tone interacts with other processes in the phonology of the phrase level. First, let's consider a simple case: /kubala H kwa:ko/ 'your (act of) counting me,' which has the surface form [kubalá$_1$ kwá$_1$:ko]. This form undergoes lapse avoidance, which associates the floating high tone, and tone spread, which spreads the singly linked high tone as discussed in the previous section.

In the first pass through the grammar, the floating high tone associates. This is due to the ranking of the markedness constraint *Float, which assigns a violation to unassociated tones (Myers 1997), above *Assoc, which allows the floating tone to dock onto a TBU. This step is shown in (132):

(132) Step 1: Tonal morpheme associates

/kubala H kwa:ko/	*Float	*Assoc
a. kubala H kwa:ko	W$_1$	L
b. → kubalá$_1$ kwa:ko		1

In this tableau, association of the floating high is a harmonically improving step.

In the first step of the derivation, association of the floating high tone is preferred over deletion. This is due to a ranking of Max-H above *Assoc, shown in the following tableau:

(133) Step 1: Association preferred over deletion

/kubala H kwa:ko/	*Float	Max-H	*Assoc
a. kubala H kwa:ko	W_1		L
b. → kubalá$_1$ kwa:ko			1
c. kubala kwa:ko		W_1	L

In this tableau, the winner has association of the floating tone to a TBU, which is preferred over deletion of the floating high tone in candidate (c).

In the previous section, the satisfaction of the ToneMin constraint was necessary for the process of tone spread at the phrase level. This constraint will also be active in the tone spread of lapse avoidance high tones. However, in this form, we do not see any critical interactions in the ordering of ToneMin, because its violation cannot be removed in a single step. The following tableau shows violations of ToneMin in the first pass through the grammar:

(134) Step 1: No satisfaction of ToneMin

/kubala H kwa:ko/	ToneMin	*Float	Max-H	*Assoc
a. kubala H kwa:ko	1	W_1		L
b. → kubalá$_1$ kwa:ko	1			1
c. kubala kwa:ko	1		W_1	L

In this tableau, each of the candidates has a violation of ToneMin. Candidate (a) has a violation because the floating high tone is not minimally linked to two TBUs, which is also true for the singly linked high tone in candidate (b). Candidate (c) has a violation due to the floating high tone in the input, despite the deleted high tone in the output. Because a floating high tone requires two steps of association (or deletion and an additional pass through the grammar) to satisfy ToneMin, it cannot be satisfied in the first pass. The initial step must be motivated by some other markedness constraint, in this case *Float.

Returning to lapse avoidance, we will need additional constraints to prevent association of the floating high tone to other TBUs. This high tone attaches to the word preceding the floating tone, so I will use a positional version of the Local constraint (Myers 1997), to prevent the high tone from associating to a non-local position. The original version is defined as follows (Myers 1997):

(135) Local: If an input tone T has an output correspondent T′, some edge of T must correspond to some edge of T′.

This constraint prevents high tones from associating to non-adjacent syllables, but we will need a positional constraint to restrict movement to an adjacent leftward syllable. This is accomplished using a positional Local-L constraint, defined as follows:

(136) Local-L: If an input tone T has an output correspondent T′, the left edge of T must correspond to some edge of T′.

For an associated high tone, this constraint will prevent association to all non-adjacent syllables, but will not restrict spreading to adjacent TBUs in either direction if the input association line is maintained.[10] In case of a floating tonal morpheme, the only TBU which does not violate this constraint is the one immediately left of the floating tone's input location, as shown in the following tableau:

[10] Which it must be in a single step harmonic grammar.

(137) Step 1: Tonal morpheme associates left of the host

/kubala H kwa:ko/	ToneMin	Local-L	*Float	Max-H	*Assoc
a. kubala H kwa:ko	1		W_1		L
b. → kubalá$_1$ kwa:ko	1				1
c. kubala kwá$_1$:ko	1	W_1			1
d. kubá$_1$la kwa:ko	1	W_1			1
e. kubala kwa:ko	1			W_1	L

Candidate (b), the winner, does incur any violations of the positional Local constraint, while candidates (c) and (d) each incur a violation due to an intervening TBU between the boundaries. The ranking of Local-L cannot be determined from this form because the losing candidates are harmonically bounded by the winner. The ranking will be determined by other interactions.

Following association of the floating high tone, the high tone spreads right via the regular process of tone spread to remove the violation of ToneMin:

(138) Step 2: Spreading of morphological tone

/ kubalá$_1$ kwa:ko/	ToneMin	Local-L	*Float	Max-H	*Assoc
a. kubalá$_1$ kwa:ko	W_1				L
b. → kubalá$_1$ kwá$_1$:ko					1

The grammar selects candidate (b), which is harmonically improving due to tone spread and the removal of the ToneMin violation.

As discussed earlier, tone spread is rightward due to the ranking *Assoc-L>>ToneMin. Rightward spread is maintained with *Assoc-L added to the tableau in (139):

(139) Step 2: Rightward spreading of morphological tone

/ kubalá$_1$ kwa:ko/	*Assoc-L	ToneMin	Local-L	*Float	Max-H	*Assoc
a. kubalá$_1$ kwa:ko		W$_1$				L
b. → kubalá$_1$ kwá$_1$:ko						1
c. kubá$_1$lá$_1$ kwa:ko	W$_1$					1

As seen in this tableau, the grammar prefers rightward to leftward spreading, selecting candidate (b) as the winner.

In the final step, the grammar converges on this form:

(140) Step 3: Converge

/kubalá$_1$ kwá$_1$:ko/	*Assoc-L	ToneMin	Local-L	*Float	Max-H	*Assoc
a. → kubalá$_1$ kwá$_1$:ko						
b. kubalá$_1$ kwá$_1$:kó$_1$						W$_1$

The fully faithful form is selected as the winner, candidate (a). Candidate (b) with an additional violation of *Associate is not harmonically improving because this violation does not satisfy any higher ranked markedness constraint. The grammar converges in this step, resulting in the surface form, [kubalá$_1$ kwá$_1$:ko].

Thus far we have not seen a crucial ranking between ToneMin and *Float. In the previous example, the grammar was able to repair first the violation of *Float, followed by the violation of ToneMin. However, in the next form, /ku:mbá$_1$la H kwa:ko/, there are two initial violations of ToneMin and one violation of *Float. This form provides evidence for the ranking between ToneMin and Float because it requires tone spread to block the association of the floating high tone, resulting in the surface form, [ku:mbá$_1$lá$_1$ kwa:ko]. This is shown in the derivations in (141):

(141) Tone spread blocks association

a. /ku:mbá$_1$la H kwa:ko/ → ku:mbá$_1$lá$_1$ H kwa:ko → [ku:mbá$_1$lá$_1$ kwa:ko]

b. /ku:mbá$_1$la H kwa:ko/ → ku:mbá$_1$lá$_2$ kwa:ko → *[ku:mbá$_1$lá$_2$ kwá$_2$:ko]

The derivation in (a) shows the correct surface form with an initial step of tone spread followed by deletion of the floating tone. It is crucial for tone spread to apply first, blocking the docking site of lapse avoidance. In contrast, (b) shows an incorrect derivation leading to the wrong surface form. If lapse avoidance is allowed to apply first, it will dock and spread, resulting in an incorrect surface form.

In a constraint-based account of these interactions, the constraint motivating tone spread must dominate that the one motivating lapse avoidance, or ToneMin>>*Float. This ranking requires the step of tone spread to apply first, as in (142):

(142) Step 1: Tone spread

/ku:mbá$_1$la H kwa:ko/	Assoc-L	ToneMin	Local-L	*Float	Max-H	*Assoc
a. ku:mbá$_1$la H kwa:ko		W$_2$		1		L
b. → ku:mbá$_1$lá$_1$ H kwa:ko		1		1		1
c. ku:mbá$_1$lá$_2$ kwa:ko		W$_2$		L		1

In this tableau, the winner is one which removes a violation of the highest ranked markedness constraint, ToneMin. The possible step of associating the floating high tone is not the most harmonically improving step.

In the following step, the violation of *Float is repaired via deletion:

(143) Step 2: Tone deletion

/ku:mbá$_1$lá$_1$ H kwa:ko/	Assoc-L	ToneMin	Local-L	*Float	Max-H	*Assoc
a. ku:mbá$_1$lá$_1$ H kwa:ko		1		W$_1$	L	
b. →ku:mbá$_1$lá$_1$ kwa:ko		1			1	
c. ku:mbá$_1$lá$_1$ kwá$_2$:ko		1	W$_1$		L	

The violation of ToneMin is due to the floating high tone, which as was seen in the previous form, cannot be repaired in a single step. Instead, the grammar will choose a winner that removes the violation of *Float. There is no possible docking site which is harmonically improving, as seen by the fatal violation of Local-L in candidate (c). Thus, the winner is candidate (b), which deletes the floating high tone.

With the final harmonically improving step, the form will converge on the next pass though the grammar, as in (144):

(144) Step 3: Converge

/ku:mbá$_1$lá$_1$ kwa:ko/	Assoc-L	ToneMin	Local-L	*Float	Max-H	*Assoc
a. → ku:mbá$_1$lá$_1$ kwa:ko						
b. ku:mbá$_1$lá$_1$ kwá$_1$:ko						W$_1$

There are no remaining markedness constraints to be removed, so all other candidates will be harmonically bounded by the winner, candidate (a). This derivation results in the final surface form [ku:mbá$_1$lá$_1$ kwa:ko].

In the final form, both tone spread and lapse avoidance occur. The input to the phrase level, /kugí$_1$tuha H kwa:ko/ will surface as [kugí$_1$tú$_1$há$_2$ kwá$_2$:ko][11]. This input has two violations of ToneMin. Unlike the previous form, the singly linked tone is on the antepenultimate, rather than the penultimate TBU. In the first step, this high tone undergoes tone spread:

(145) Step 1: Tone spread

/kugí$_1$tuha H kwa:ko/	Assoc-L	ToneMin	Local-L	*Float	Max-H	*Assoc
a. kugí$_1$tuha H kwa:ko		W$_2$		1		L
b. → kugí$_1$tú$_1$ha H kwa:ko		1		1		1
c. kugí$_1$tuhá$_2$ kwa:ko		W$_2$				1

The winner, candidate (b), removes a violation of ToneMin by associating to the following TBU. This is preferred over removing the violation of the lower ranked *Float,

[11] The actual surface form is [kugí$_1$tú$_1$ꜜhá$_2$ kwá$_2$:ko], with an additional specification of downstep. Like Odden (2008), we will attribute this downstep to post-phonological phonetic implementation.

as in candidate (c). It is important to note that this step of tone spread does not block the association site of the floating high tone.

In the next step, the grammar will select the form which repairs the violation of *Float. In this case, association of the floating high tone is not blocked by the previous step of tone spread, so tone association can apply:

(146) Step 2: Tone association

/kugí$_1$tú$_1$ha H kwa:ko/	Assoc-L	ToneMin	Local-L	*Float	Max-H	*Assoc
a. kugí$_1$tú$_1$ha H kwa:ko		1		W$_1$		L
b. → kugí$_1$tú$_1$há$_2$ kwa:ko		1				1
c. kugí$_1$tú$_1$ha kwa:ko		1			W$_1$	L

The winner, candidate (b), removes the violation of *Float by associating to the adjacent TBU. This is preferred over candidate (c), which repairs *Float via the higher ranked Max-H.

With this step of tone association, a new sequence of consecutive high tones is created. The constraint *HH is not ranked high enough to block the step of tone association. This is due to the crucial ranking of Max-H>>*HH, which is shown in tableau (147):

(147) Step 2: Tone association can create consecutive high tones.

/kugí$_1$tú$_1$ha H kwa:ko/	Assoc-L	ToneMin	Local-L	*Float	Max-H	*Assoc	*HH
a. kugí$_1$tú$_1$ha H kwa:ko		1		W$_1$		L	L
b. → kugí$_1$tú$_1$há$_2$ kwa:ko		1				1	1
c. kugí$_1$tú$_1$ha kwa:ko		1			W$_1$	L	L

In this tableau, the ranking of the *HH constraint does not affect the selection of the winner, candidate (b).

In the next step, the remaining violation of ToneMin is removed by tone spread:

(148) Step 3: Tone spread

/kugí$_1$tú$_1$há$_2$ kwa:ko/	Assoc-L	ToneMin	Local-L	Float	Max-H	*Assoc	*HH
a. kugí$_1$tú$_1$há$_2$ kwa:ko		W$_1$				L	L
b. → kugí$_1$tú$_1$há$_2$ kwá$_2$:ko						1	1

Candidate (b) is selected as the winner, removing the final markedness constraint violation. In the next pass through the grammar, the derivation will converge, as in (149):

(149) Step 4: Converge

/ kugí$_1$tú$_1$há$_2$ kwá$_2$:ko /	Assoc-L	ToneMin	Local-L	Float	Max-H	*Assoc	*HH
a. → kugí$_1$tú$_1$há$_2$ kwá$_2$:ko							1
b. kugí$_1$tú$_1$ha kwá$_2$:ko		W$_1$					L

In this final step, the grammar converges on the surface form [kugí$_1$tú$_1$há$_2$ kwá$_2$:ko]. Any additional steps to repair the violation of *HH will not be harmonically improving, as they will violate the higher ranked constraint ToneMin.

With these forms, the final ranking for the phrase level grammar is as follows:

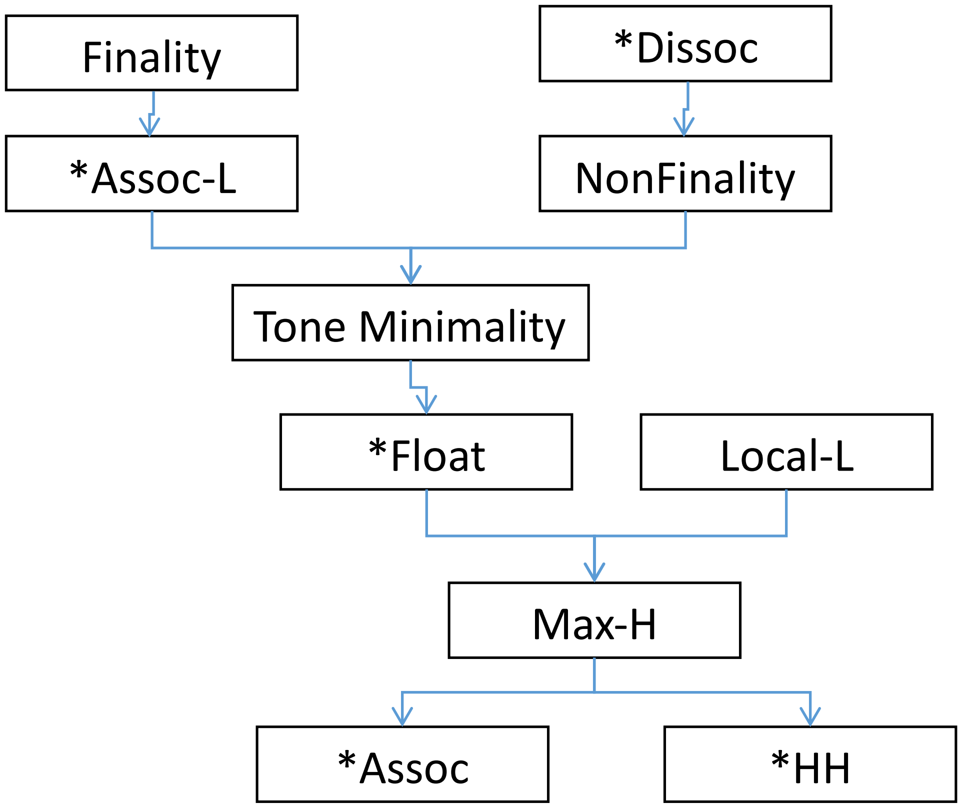

Figure 5: Final phrase level ranking

Given these rankings, I will review the derivations for some of the critical phrase level forms.

First, I will show the derivation for /kubala H kwa:ko/, which shows lapse avoidance and tone spread, in (150)-(152):

(150) Step 1

/ kubala H kwa:ko /	Fin Min	*Diss	Non Fin	*Assoc-L	Tone Min	Local-L	*Float	Max -H	*Assoc	*HH
a. kubala H kwa:ko					1		W$_1$		L	
b. → kubalá$_1$ kwa:ko					1				1	
c. kubala kwa:ko					1			W$_1$	L	
d. kubala kwá$_1$:ko					1	W$_1$			1	

(151) Step 2

/ kubalá$_1$ kwa:ko /	Fin Min	*Diss	Non Fin	*Assoc-L	Tone Min	Local-L	*Float	Max -H	*Assoc	*HH
a. kubalá$_1$ kwa:ko					W$_1$				L	
b. → kubalá$_1$ kwá$_1$:ko									1	
c. kubá$_1$lá$_1$ kwa:ko				W$_1$						

(152) Step 3: Converge

/ kubalá$_1$ kwá$_1$:ko /	Fin Min	*Diss	Non Fin	*Assoc-L	Tone Min	Local-L	*Float	Max -H	*Assoc	*HH
a. → kubalá$_1$ kwá$_1$:ko										
b. kubalá$_1$ kwá$_1$:kó$_1$			W$_1$						W$_1$	
c. kubalá$_1$ kwa:ko		W$_1$			W$_1$					

The first step of the derivation shows association of the floating high tone to the locally leftward TBU. In the following step, the high tone spreads rightward to repair the violation of ToneMin. In the final step the derivation converges on [kubalá$_1$ kwá$_1$:ko].

The next derivation is of the form /ku:mbá$_1$la H kwa:ko/. This form undergoes tone spread, followed by floating tone deletion, shown in (153)-(155):

(153) Step 1

/ ku:mbá$_1$la H kwa:ko /	Fin Min	*Dis	Non Fin	*Asso -L	Tone Min	Local- L	*Float	Max -H	*Assoc	*HH
a. ku:mbá$_1$la H kwa:ko					W$_2$		1		L	
b.→ku:mbá$_1$lá$_1$ H kwa:ko					1		1		1	
c. ku:mbá$_1$la kwa:ko					W$_2$			W$_1$	L	
d. ku:mbá$_1$lá$_2$ kwa:ko					W$_2$				1	W$_1$

(154) Step 2

/ ku:mbá$_1$lá$_1$ H kwa:ko /	Fin Min	*Dis	Non Fin	*Asso -L	Tone Min	Local- L	*Float	Max -H	*Assoc	*HH
a. ku:mbá$_1$lá$_1$ H kwa:ko					1		W$_1$	L		
b.→ku:mbá$_1$lá$_1$ kwa:ko					1			1		
c. ku:mbá$_1$lá$_1$ kwá$_2$:ko					1	W$_1$		L	W$_1$	W$_1$

(155) Step 3: Converge

/ ku:mbá$_1$lá$_1$ kwa:ko /	Fin Min	*Dis	Non Fin	*Asso -L	Tone Min	Local- L	*Float	Max -H	*Assoc	*HH
a. → ku:mbá$_1$lá$_1$ kwa:ko										
b.ku:mbá$_1$lá$_1$ kwá$_1$:ko									W$_1$	

In this derivation, the singly linked high tone spreads, blocking lapse avoidance. In the next step, the floating high tone deletes, followed by convergence on [ku:mbá$_1$lá$_1$ kwa:ko].

In the final derivation, I will show /kugí$_1$tuha H kwa:ko/. This form undergoes tone spread, followed by lapse avoidance, followed by tone spread again. This derivation is shown in (156)-(159):

(156) Step 1

/ kugí$_1$tuha H kwa:ko /	Fin Min	*Dis	Non Fin	*Asso -L	Tone Min	Local-L	*Float	Max -H	*Assoc	*HH
a. kugí$_1$tuha H kwa:ko					W$_2$		1		L	
b.→ kugí$_1$tú$_1$ha H kwa:ko					1		1		1	
c. kugí$_1$tuhá$_1$ kwa:ko					W$_2$		L			
d. kugí$_1$tuha kwa:ko					W$_2$		L	W$_1$		

(157) Step 2

/ kugí$_1$tú$_1$ha H kwa:ko /	Fin Min	*Dis	Non Fin	*Asso -L	Tone Min	Local-L	*Float	Max -H	*Assoc	*HH
a. kugí$_1$tú$_1$ha H kwa:ko					1		W$_1$		L	L
b.→ kugí$_1$tú$_1$há$_2$ kwa:ko					1				1	1
c. kugí$_1$tú$_1$ha kwá$_2$:ko					1	W$_1$			1	L
d. kugí$_1$tú$_1$ha kwa:ko					1			W$_1$	L	L

(158) Step 3

/ kugí$_1$tú$_1$há$_2$ kwa:ko /	Fin Min	*Dis	Non Fin	*Asso -L	Tone Min	Local-L	*Float	Max -H	*Assoc	*HH
a. kugí$_1$tú$_1$há$_2$ kwa:ko					W$_1$				L	1
b.→ kugí$_1$tú$_1$há$_2$ kwá$_2$:ko									1	1

(159) Step 4: Converge

/ kugí$_1$tú$_1$há$_2$ kwá$_2$:ko /	Fin Min	*Dis	Non Fin	*Asso -L	Tone Min	Local-L	*Float	Max -H	*Assoc	*HH
a. → kugí$_1$tú$_1$há$_2$ kwá$_2$:ko										1
b.kugí$_1$tú$_1$ha kwá$_2$:ko		W$_1$			W$_1$					L

In the first step of the derivation, the singly linked high tone spreads. In the next step, the floating high tone associates to the left local TBU. Next, the second singly linked high tone spreads. Finally, the derivation converges on [kugí$_1$tú$_1$há$_2$ kwá$_2$:ko].

The phrase level is the final strata of the grammar. These forms will be the final outputs of the phonology.

2.3 Advantages over other approaches

In the previous section, I have provided a Stratal Harmonic Serialism analysis of tonal opacity in Kikerewe. This approach has advantages over the related constraint-based frameworks of parallel OT, Stratal OT, and Harmonic Serialism. An initial argument against an OT analysis comes from Odden (2000), who argues for a rule-based approach, due to the limitations of parallel OT in accounting for this data.

One issue raised by Odden (2000) is the difficulty of accounting for directional deletion of consecutive high tones in parallel OT. An illustration of the issue is shown in tableau (160), with the input /abatá$_1$:tú$_2$gí$_3$ká$_4$:la:ngi:zye/ containing four consecutive high tones that need to be reduced due to the ranking of *HH>>Max-H. This example also considers the *Tone Minimality constraint, which prefers doubly-linked tones.

(160) Consecutive high tone deletion in OT (Odden 2000)

/abatá$_1$:tú$_2$gí$_3$ká$_4$:la:ngi:zye/ | | | | H H H H	*HH	*Tone Minimality	Max-H
a. abatá$_1$:tú$_2$gí$_3$ká$_4$:lá$_4$:ngi:zye | | | V H H H H	*!**	***	
b. abatá$_1$:tugika:la:ngi:zye | H		*!	***
c. → abatá$_1$:tú$_1$gika:la:ngi:zye V H			**!*
d. X abatá$_1$:tú$_1$giká$_2$:lá$_2$:ngi:zye V V H H			**

The desired winner in this example is candidate (c), [abatá$_1$:tú$_1$gika:la:ngi:zye], which exhibits deletion of all but the leftmost high tone, which also rightward to the adjacent TBU. Instead, this grammar incorrectly selects candidate (d), [abatá$_1$:tú$_1$giká$_2$:lá$_2$:ngi:zye], which minimizes violations of faithfulness constraint Max-H, while also removing all violations of the markedness constraints. This pattern is reminiscent of the pathological global optima discussed in the Harmonic Serialism literature (McCarthy 2008, 2009). The globally optimal pattern deletes every third H in order to satisfy *HH and *Tone Minimality simultaneously, leaving a low toned TBU between doubly-linked high tones. Odden's solution is to propose a two-level constraint (Koskenniemi 1983; Karttunen 1993), */H/H. There are arguments against the use of two-level constraints, such as the lack of generalizations compared to rule-based accounts and traditional OT constraints (McCarthy 1996). In this case, the two-level constraint */H/H combines the general dispreference for HH sequences and the preferred direction of repair into a single constraint. Though there exist cases of local constraint conjunction of markedness and faithfulness constraints (Lubowicz 2002; Baković 1999; Morris 2002), there is an absence of Max and Markedness conjunction (Moreton and Smolensky 2002). The current proposal in Stratal Harmonic Serialism maintains the *HH markedness constraint and proposes an additional positional faithfulness constraint *Dissoc-L, rather than using a two-level constraint. The mechanisms of the framework play a major role in the use of these constraints. Crucially, the gradualness property of Harmonic Serialism limits the application of multiple simultaneous operations that motivate the use of a two-level approach in Odden's analysis; namely, if multiple steps of tone dissociation can occur simultaneously, the input needs to be referenced in order to establish directionality. Because Harmonic Serialism limits Gen to a single operation per step of the grammar, the additional reference is no longer needed.

A second issue raised by Odden (2000) is the failure of OT to handle the full range of tonal interaction in Kikerewe, even with the use of */H/H. While underlying consecutive high tone sequences are deleted, those created through other processes can surface. Consider the two cases in (161), where form (a) has underlying consecutive high tones and no surface consecutive high tones, while form (b) has no underlying consecutive high tone and two surface consecutive high tones.

(161) a. /abatá$_1$ː-tú$_2$-gí$_3$-kú$_4$-hé$_5$ːleːzye/ → [abatá$_1$ːtú$_1$gikuheːleːzye]

b. /á$_1$-ku-chú$_2$mita/ → [akú$_1$chú$_2$mí$_2$ta]

The use of */H/H to handle (161)a has already been been discussed, but it also handles (161)b because the consecutive high tones in the output were not consecutive underlyingly, thus they do not violate */H/H, but would have violated *HH. However, some derived consecutive high tones are prohibited, and the two-level account does not handle these cases, as shown in (162).

(162) a. /ku-bala kwaːko/ → [kubalá$_1$ kwá$_1$ːko]

b. /kuː-ḿ$_1$-bala kwaːko/ → [kuːmbá$_1$lá$_1$ kwaːko] *[kuːmbá$_1$lá$_2$ kwá$_2$ːko]

The form in (162)a shows the process of lapse avoidance, in which a high tone is inserted on the final syllable and spreads rightward. In (162)b, insertion of a floating high tone is blocked due to MR, though the sequence is derived, through both the insertion of the high tone and shift of the tone from the nasal modifier. Odden (2000) acknowledges that this is a limitation of the approach, suggesting that a stratal approach would improve the analysis of this data. My analysis of Kikerewe in Stratal Harmonic Serialism does include this component, proposing different constraint rerankings between levels.

2.4 Conclusion

In this case study, I have provided an analysis of tonal opacity in Kikerewe using the framework of Stratal Harmonic Serialism. The use of this framework has benefits over other approaches, which cannot account for the data or require the use of additional mechanisms, such as two-level constraints. Under this analysis, both the Harmonic Serialism and Stratal OT components are motivated at different stages of the analysis. Harmonic Serialism is useful in correctly deriving directional consecutive high tone deletion, while the Stratal OT component allows for reranking of the constraints governing opaque high tone deletion between levels.

This analysis proposes the use of a Stratal Harmonic Serialism framework in order to fully account for the data, but the predictive power of the framework is not entirely known. In chapters 3 and 4, I investigate the interactions between the Harmonic Serialism and Stratal OT components of this framework, exploring the typological and opaque predictions made by the model.

Chapter 3 Syllable Structure Typology: Restrictiveness

In this chapter, I present a typology of syllable structure using Gradual Syllabification in Stratal Harmonic Serialism, a theory that gradually builds syllables in harmonically improving steps. The examination of a complete factorial typology is useful for exploring the generative capacity of the framework, which should be not be undergenerative or pathologically overgenerative. Syllable structure is a useful domain for exploring the generative capacity of a constraint-based framework, beginning with the early work on typologies in Optimality Theory. Because Harmonic Serialism is more restrictive than parallel OT, the set of typological predictions can be used to assess whether the framework is undergenerative. Similarly, the Stratal component of the framework makes it less restrictive than a single-level grammar, so the typology can be used to assess whether the framework is overgenerative. Using the typological predictions generated here, I show that the framework neither overgenerates nor undergenerates in the domain of syllable structure.

The domain of syllable structure is especially relevant for Stratal Harmonic Serialism because the gradual nature of the framework, both within and between levels, is particularly illuminating for interactions involving phonological structure. In presenting this typology, a number of interesting observations about the nature of gradual structure building and locally optimal derivations arise. The preference for locally-optimal over globally-optimal derivations has been shown to be a beneficial property of Harmonic Serialism (McCarthy 2008, 2009; Pruitt 2010), but in the domain of syllable structure this can lead to syllable type inventories in violation of implicational markedness under some constraint and operation definitions. While being particularly important for Stratal

Harmonic Serialism, these results also have implications for Harmonic Serialism more generally, shedding light on novel types of interactions.

This chapter is presented in five parts. In section 3.1, I provide an outline of the model used to generate the typology of syllabification. In section 3.2, I present results of the model, which attempt to answer core questions about the framework, showing that it is capable of generating a reasonable typology. In section 3.3, I present alternate simulations, changing the results of some crucial assumptions made in previous sections. In section 3.4, I discuss some implications of the simplifying assumptions. In section 3.5, I discuss the implications of these results for Stratal Harmonic Serialism and Harmonic Serialism.

3.1 Gradual Syllabification in Stratal Harmonic Serialism

In this section, I outline the specifics of Gradual Syllabification in Stratal Harmonic Serialism with regard to the structure, constraints, and operations that are used in computational simulations to generate a syllable structure typology. The specific assumptions made by the model are crucial to the resulting typology because even small changes have the potential for significant effects. Some of these assumptions diverge from other models, with the goal of achieving a final typology that is neither overgenerative nor undergenerative.

The gradual derivation of syllable structure in Optimality Theory was first introduced by Prince and Smolensky (1993/2004), who demonstrated the mechanisms of serial harmonic syllabification with an analysis of Tashlhiyt Berber. Though much of the work on syllable structure in OT has focused on the parallel variety, recent work has provided accounts of syllabification in serial grammars. A proposal by Elfner (Elfner

2009) lays out the constraints and operations that could be used for a model of serial syllabification in Harmonic Serialism, with the goal of accounting for opaque stress-epenthesis interactions. In this analysis, syllables are built gradually, parsing segments in separate operations until convergence. The building of syllables interacts with other relevant operations, like stress assignment, epenthesis, and resyllabification. Elfner's proposal provides a starting point for understanding how constraints and operations interact in a serial syllabification model. While the current proposal maintains many of the same basic constraints and operations from Elfner's account, there are some significant deviations due to the different structure of the model and different goals.

First, a crucial component of the Stratal Harmonic Serialism model is the interaction between different levels of the grammar. This results in different constraint rankings between levels, as well as the introduction of affixal material at later levels. These properties require different mechanisms than a serial model that only operates at a single level. Second, the focus of the current proposal is computing a syllable structure typology. In doing so, previously undiscovered issues arise, necessitating a reworking of some constraints and operations. The unique needs of Gradual Syllabification in Stratal Harmonic Serialism require changes from earlier proposals. In the remainder of this section, I will lay out the exact details of the constraints and operations used in the current proposal.

3.1.1 Structure of the model

The structure of the computational model includes the basic components of Stratal Harmonic Serialism.[12] A grammar is a set of constraint rankings for each level of the grammar, with two levels being considered here.

For each unique grammar of two strict constraint rankings, a set of input-output mappings are determined. The set of inputs are consistent across grammars. Outputs are calculated through an iterative two-step process. First, GEN uses the input to determine a set of possible candidates based on the set of operations. Next, EVAL selects a winner from the set of candidates and the given constraint ranking. If this winner is the same as the input to the grammar, convergence is reached and it is selected as the output. If not, this winner is passed through the grammar as the new input, cycling until convergence. In a Harmonic Serialism model, this would be the final output, but in Stratal Harmonic Serialism there are multiple levels rather than a single level. The output of the first level becomes the input to the second level, where the process is repeated. At the final level, the output is the final output of the grammar. This process is summarized in Figure 6.

[12] Code excerpts are provided in Appendix.

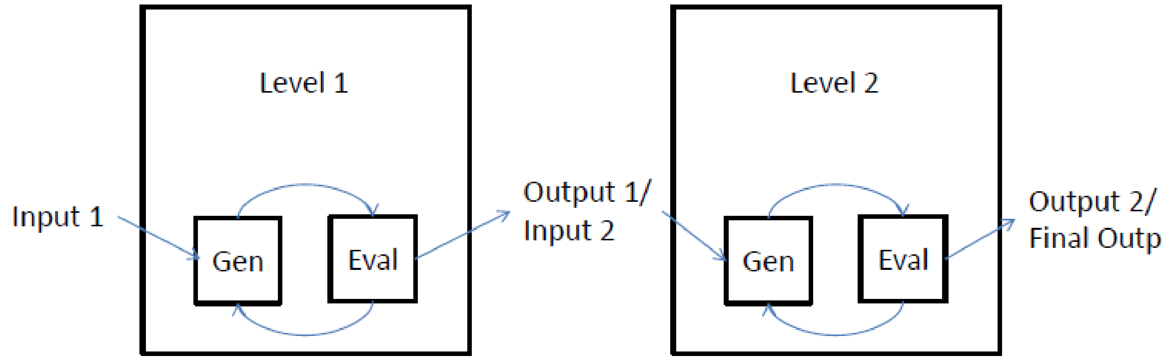

Figure 6: Structure of the model

Each level operates independently, with its own constraint ranking, input, and output. Outputs from the first level are passed as the input to the following level, with the potential for affixal material to be added between levels.

The structure described here differs considerably from other models of Harmonic Serialism. One effect of having multiple levels of the grammar is that the set of inputs to the first level is not the same as the set of inputs to later levels. Inputs to the first level most closely resemble inputs in other single-level grammars, with no structure in underlying forms. Second level inputs differ considerably because any structure built in the first level is maintained in inputs to the second level. Additionally, second level inputs can contain affixal material, which do not have any underlying structure. With regard to syllabification, inputs to the first level are all unsyllabified, while inputs to the second level include a mix of syllabified inputs, which were the outputs of the first level, and partially syllabified inputs, which are the outputs of the first level concatenated with unsyllabified affixal material. Any unparsed segments in the final output of a level are removed, as in stray erasure (McCarthy 1979; Steriade 1982; Ito 1986, 1989). Stray erasure in the model is a simplifying assumption, but is not expected to have a significant effect between levels, due to the overlap with the unparsed affixal material added to inputs. While the inputs to the second level contain both syllabified and unsyllabified material, all unsyllabified material is affixal; stray erasure removes any remaining unparsed segments between levels. Furthermore, this is different from the traditional HS interpretation of all candidates being syllabified automatically, rather than undergoing separate steps of syllabification. A given constraint ranking will predict different mappings depending on whether the input is syllabified or not, so this distinction is crucial.

A computational model of Stratal Harmonic Serialism containing two levels of the grammar is able to capture all the important properties that are under consideration. The model is able to test the effect of having a constraint re-ranking between levels, as well as the effect of adding affixal material between levels. While the theoretical model posits three levels (stem, word, and phrase), the absence of a third level in this analysis should not be cause for concern. We can expect that the generalizations that hold between levels one and two will also hold between levels two and three, due to the shared properties in those interlevel interactions; the typologies generated from levels one and two include the interactions between parsed and unparsed material, addition of affixes, and constraint rerankings, and these changes would be the same in subsequent levels. As I will show that the interaction between two levels is restrictive, we can expect that the interaction with a third (or fourth or fifth) level would be restrictive as well.

3.1.2 Constraints

In this section, I outline the constraints that are used in the model. There is an established tradition of syllable structure constraints in parallel OT. This includes the markedness constraints NoCoda, Onset, and *Complex, which are all necessary for a complete syllable structure analysis. There are also the faithfulness constraints, Max and Dep, which penalize segment deletion and insertion. These constraints are now standard in most constraint-based analyses of syllable structure.

For this analysis using Gradual Syllabification, an additional constraint is needed to motivate parsing of syllables. Elfner (2009) uses the constraint ParseSeg, which assigns a violation to each segment that is unparsed. Violations are removed when a segment is parsed into a syllable. I adopt this convention, using ParseSeg in the constraint set.

An additional constraint that penalizes resyllabification is needed for this analysis. While Elfner (2009) permits resyllabification as an operation, there is no constraint assigning violations for resyllabification. This constraint is necessary for the current proposal because of the role of resyllabification across levels of the grammar. Unlike a single-level model, resyllabification needs to be more restricted when considering multiple prosodic levels. For this reason, I propose a faithfulness constraint, IdSyl, that assigns violations when segments resyllabify between existing syllables.

In this analysis, I aim to use as many of these constraints as necessary for a complete analysis, while also taking into consideration computational tractability. The search space for the full range of grammars with two levels in Stratal Harmonic Serialism is $n!^2$, which becomes intractable very quickly. The difference between six and seven constraints has a very large effect on the computation time for running of the simulations. Not all the traditional constraints are needed to demonstrate the properties of the model in the typology. The interaction of faithfulness and markedness constraints can be demonstrated using only one of the major faithfulness constraints. I have pared down the constraint set to use the faithfulness constraint Max, but not Dep. With the omission of Dep, we still expect to have a representative typology, but with different mappings. When applicable, marked structures will be repaired by deletion, but not by epenthesis. Similarly, the constraint *Complex, which penalizes both complex onsets and complex codas, is used in place of the two separate constraints *Complex-Onset and *Complex-Coda. Using the constraint *Complex allows us to investigate the interaction of complex clusters with other constraints, without vastly increasing the computational cost.

The current simulations will use the following six constraints: NoCoda, Onset, *Complex, Max, ParseSeg, and IdSyl. This specific constraint set is not a crucial component of the model, but a means to find the closest approximation to the full typology without sacrificing any of the necessary constraints. Future simulations could re-evaluate the constraint set. A major deviation from Elfner (2009) is the omission of minor syllables, syllables that are non-moraic, and the corresponding markedness constraint militating against them. This possibility is discussed in section 3.4.1. Another possible alteration to the constraint set would be adding Dep back, as is investigated in section 3.4.2.

The table in (163) lists the constraints used in these simulations and gives examples of their violation profiles. The first column lists the name of the constraint, the second column lists a sample candidate, or candidate mapping for faithfulness constraints, and the third column lists the number of violations incurred by the constraint for the given candidate. The notation to indicate parsing is used throughout, with uppercase CV indicating unparsed segments, lowercase cv indicating parsed segments, and parentheses (cv) marking syllable boundaries. Any remaining unparsed segments in outputs are removed through stray erasure, and thus not considered in syllable inventories.

(163) Constraint definitions

Constraint	Candidate	Violations
NoCoda	CVC	0
	(cvc)	1
	(cvcc)	2
Onset	V	0
	(v)	1
*Complex	CCV	0
	(ccv)	1
	(cvcc)	1
ParseSeg	(cv)	0
	(cv)C	1
	CV	2
Max	(cv)C→(cv)	1
	(cvc)→(cv)	1
IdSyl	(cvc)(v)→(cv)(cv)	1

The difference between parsed and unparsed segments should be noted, particularly with regard to the syllable structure markedness constraints, which are only violated by parsed segments. For example, the unparsed string CVC does not violate NoCoda because there is no syllable containing a coda. The parsed string (cvc) does violate NoCoda because there is a syllable containing a coda.

In addition to selection of constraints, the exact violation profiles of these constraints also play an important role. In particular, the definition of NoCoda used in these simulations is different from the traditional one, in which a violation is assigned for each syllable that contains a coda. Instead, the version of NoCoda used in these simulations assigns a violation to each consonant that appears in a coda. This difference is seen in the string (cvcc), would only incur a single violation under the traditional definition, but incurs two violations in the constraint definition employed here. The reason for this change in definition is that the traditional NoCoda constraint in this context produces problematic typologies caused by gradual derivations terminating in locally optimal forms. Specifically, it produces typologies that contain syllables with complex codas, but no simple codas. Reformulating the NoCoda constraint allows intermediate forms to be harmonically improving where they are not with the traditional NoCoda constraint. Derivations showing the different predictions of the two formulations of NoCoda are presented in section 3.3.1.

3.1.3 Operations

The set of possible operations determine what candidates are generated by GEN for evaluation. Each operation occurs in a single step and only one operation can be applied in a single step. In this model, I have specified a set of operations based on theoretical considerations.

The first type of operation is parsing operations. Core syllabification parses a cv sequence into a core syllable. This is the only operation in which two segments can be parsed at once, with the remaining operations parsing one segment at a time. In addition to core syllabification, a new syllable can be created through syllable projection, in which a vowel is projected into a syllable. Syllable adjunction parses an additional segment into an existing syllable, either into an onset or a coda, simple or complex. The table in (164) summarizes the set of possible parsing operations that are used in these simulations, and the violation profile of each of those operations (where plus indicates a violation has been added and minus indicates a new violation has been removed).

(164) Parsing operations

Operation	Input	Output	Violations
Core syllabification	CV	(cv)	ParseSeg(-2)
Project major syllable	V	(v)	ParseSeg(-1),Onset(+1)
Onset adjunction	C(v)	(cv)	ParseSeg(-1),Onset(-1)
Coda adjunction	(cv)C	(cvc)	ParseSeg(-1),NoCoda(+1)
Complex onset adjunction	C(cv)	(ccv)	ParseSeg(-1),Complex(+1)
Complex coda adjunction	(cvc)C	(cvcc)	ParseSeg(-1),NoCoda(+1),Complex(+1)

As noted, the parsing of minor syllables is not a possible operation in this analysis, due to undesired typologies they can predict. Extensive discussion of this issue is found at the end of the chapter.

Another operation type is segment deletion, as would be penalized by the Max constraint. This operation permits the deletion of a single consonant or vowel in each step. The corresponding operation of segment insertion is not included in the set of operations, due to the omission of Dep from the constraint set, as explained in the previous section. This is not a crucial decision for the theory, but rather a simplifying assumption to restrict the domain for the current analysis. Future analyses and simulations may include segment insertion as an operation, as is discussed in section 3.4.2. These operations are summarized in (165).

(165) Deletion operations

Operation	Input	Output	Violations
Coda deletion	(cvc)	(cv)	NoCoda(-1),Max(+1)
Onset deletion	(cv)	(v)	Onset(+1),Max(+1)
Complex coda reduction	(cvcc)	(cvc)	NoCoda(-1),*Complex(-1),Max(+1)
Complex onset reduction	(ccv)	(cv)	*Complex(-1),Max(+1)
Unparsed consonant deletion	C(cv)	(cv)	ParseSeg(-1),Max(+1)
Unparsed vowel deletion	V(cv)	(cv)	ParseSeg(-1),Max(+1)

The set of deletion operations permits the deletion of one segment per step, where a segment can be a consonant or vowel, unparsed or unparsed. One potential operation missing from the set of deletion operations is the deletion of a parsed vowel. This is due to problematic interactions that can result from this operation. A full discussion of the issue is found in section 3.3.2.

The final type of operation is resyllabification, which violates the faithfulness constraint IdSyl. Resyllabification operations permit a consonant to lose its association with its current syllable and associate with an adjacent syllable in a single operation.[13] A consonant at the edge of an existing syllable can move into the edge of an adjacent syllable, realized as a coda consonant moving into the onset of the following syllable or

[13] While resyllabification is a one-step process in these simulations, it is not a crucial assumption of the theory, though the effect on the typology is not explored here. Resyllabification is reconsidered as a two-step process in the Opacity chapter.

an onset consonant moving into the coda of the preceding syllable. The resyllabification operations are summarized in (166).

(166) Resyllabification operations

Operation	Input	Output	Violations
Onset resyllabification	(v)(cv)	(vc)(v)	NoCoda(+1),Onset(+1),IdSyl(+1)
Coda resyllabification	(vc)(v)	(v)(cv)	NoCoda(-1),Onset(-1),IdSyl(+1)
Complex onset resyllab.	(v)(ccv)	(vc)(cv)	*Complex(-1),NoCoda(+1),IdSyl(+1)
Complex coda resyllab.	(vcc)(v)	(vc)(cv)	*Complex(-1),NoCoda(-1),Onset(-1), IdSyl(+1)

This section summarizes the complete set of operations that are permitted by the model. Using the constraints and operations as described here, the model generated a set of all possible syllable inventories predicted by the model. These results are presented in the next section.

3.1.4 Interaction of constraints and operations

Based on the constraints and operations described in the preceding sections, I will show some derivations demonstrating how these constraints and operations interact in Stratal Harmonic Serialism.

3.1.4.1 Parsing at the first level

First, I will demonstrate the basics of gradually parsing a syllable. Consider the ranking ParseSeg>>Max>>NoCoda>>Onset>>*Complex>>IdSyl with the input /CVC/. In a serial framework like Stratal Harmonic Serialism, the derivation occurs in multiple passes through the grammar, rather than a single pass. With the high ranking of ParseSeg, the optimal first step is core syllabification, in which a core (cv) syllable is parsed in a single operation, shown in tableau (167).

(167) Step 1: Core syllabification

/CVC/	ParseSeg	Max	NoCoda	Onset	*Complex	IdSyl
a. CVC	W3					
b. →(cv)C	1					
c. C(v)C	W2			W1		
d. CV	W2	W1				

Candidate (a) is fully faithful, and incurs three violations of ParseSeg, one for each unparsed segment. Candidate (b), the winner, undergoes core syllabification, parsing a (cv) syllable and leaving a single remaining unparsed segment, which incurs one violation of ParseSeg. Core syllabification is the only operation that permits the parsing of two segments in one step. Also, note that IdSyl is not violated by the parsing as segments, as it only assigns violations when a segment changes an existing syllable association, or resyllabification. Candidate (c) undergoes the syllable projection operation, parsing only a single segment. This leaves two remaining unparsed segments, incurring two violations of ParseSeg. Additionally, candidate (c) parses a (v) syllable, incurring a violation of Onset. Candidate (d) deletes a segment, removing a violation of ParseSeg, but incurring a violation of Max.

Candidate (b) is the winner here because it minimizes the violations of the highly ranked ParseSeg. For the initial step of parsing, if there is a CV string in the input, core syllabification will always be preferred over the parsing of a single segment because it removes two violations of ParseSeg, rather than one. As seen in tableau (167), the core syllabification candidate (b) harmonically bounds the syllable projection candidate (c). As the winner, candidate (b) becomes the input of step 2, shown in tableau (168).

(168) Step 2: Coda adjunction

/(cv)C/	ParseSeg	Max	NoCoda	Onset	*Complex	IdSyl
a. (cv)C	W1		L			
b. →(cvc)			1			
c. (cv)		W1	L			

In this step, candidate (a) is faithful, incurring a violation of ParseSeg for the remaining unparsed C segment. Candidate (b) undergoes the coda adjunction operation, losing a violation of ParseSeg and adding a violation of the lower ranked NoCoda. Candidate (c) undergoes the deletion operation by deleting the remaining unparsed C, thus losing a violation of ParseSeg and adding a violation of Max. Candidate (b) wins because the other candidates fatally violate higher ranked constraints, and it becomes the input to step 3 in (169).

(169) Step 3: Converge

/(cvc)/	ParseSeg	Max	NoCoda	Onset	*Complex	IdSyl
a. →(cvc)			1			
b. (cv)		W1	L			

In this step, candidate (a) is the faithful candidate, with one violation of NoCoda. Candidate (b) undergoes the deletion operation, deleting the existing coda and incurring a violation of Max. Due to the ranking Max>>NoCoda, candidate (a) wins. When the faithful candidate wins, there are no remaining harmonically improving operations, and the derivation converges.

3.1.4.2 Parsing at the second level

Now that I have demonstrated the basics of syllable parsing at the first level, let us examine the differences at the second level. First, the inputs are no longer the unparsed strings seen at the first level, but the outputs of the first level with the possibility of unparsed affixal material. Let us consider one such input /(cvc)-V/, which contains the output (cvc) from the previous level and a V affix. Second, the constraints can be re-ranked between levels. Let us consider a minimally different ranking, ParseSeg>>Max>>NoCoda>>*Complex>>Onset>>IdSyl, where the relative order of Onset and *Complex has been reversed from the previous level. The first step for this derivation at the second level is shown in tableau (170).

(170) Step 1: Syllable projection

/(cvc)-V/	ParseSeg	Max	NoCoda	*Complex	Onset	IdSyl
a. (cvc)V	W1		1		L	
b. →(cvc)(v)			1		1	
c. (cvc)		W1	1		L	

In this step, candidate (a) is the faithful candidate, with violations of ParseSeg and NoCoda. Candidate (b) undergoes syllable projection, parsing the V affix into an onsetless syllable, losing a violation of ParseSeg and incurring a violation of Onset. Candidate (c) undergoes deletion, deleting the unparsed V affix, losing a violation of ParseSeg and incurring a violation of Max. Candidate (b) wins, with the other candidates fatally violating ParseSeg and Max, and becomes the input to Step 2 in (171).

(171) Step 2: Resyllabification

/(cvc)(v)/	ParseSeg	Max	NoCoda	*Complex	Onset	IdSyl
a. (cvc)(v)			W1		W1	L
b. →(cv)(cv)						1

In this step, candidate (a) is the faithful candidate, with violations of NoCoda and Onset. Candidate (b) undergoes resyllabification, with the coda of the first syllable moving into the onset of the second syllable. With this operation, candidate (b) incurs a violation of IdSyl and repairs its violations of NoCoda and Onset. The step of resyllabification is harmonically improving, so candidate (b) is the winner over the faithful candidate, and becomes the input to Step 3 in (172).

(172) Step 3: Converge

/(cv)(cv)/	ParseSeg	Max	NoCoda	*Complex	Onset	IdSyl
a. →(cv)(cv)						
b. (cvc)(v)			W1		W1	W1

In this step, the faithful candidate has no violations. We can consider the resyllabification candidate (b), but this step is not harmonically improving. With no remaining harmonically improving operations, the derivation converges with (cv)(cv) as the final output.

Thus far, I have demonstrated how the Gradual Syllabification constraints and operations interact within the structure of Stratal Harmonic Serialism model. These mechanisms are used to generate a syllable structure typology, which is presented in the next section.

3.2 Results

In this section, I present the results of the model at a single level of the grammar, two levels of the grammar, and two levels of the grammar with affixes. The two-level typology was generated twice, testing a different set of inputs; one typology was generated with a set of simple inputs equal to the set of nine basic syllable types (section 3.2.1) and a second typology was generated with a set of rich inputs (section 3.2.2).

3.2.1 Basic inputs

In this section, I provide the results of a simulation with a set of inputs equal to the nine basic syllable types: CV, CVC, V, VC, CCV, VCC, CCVC, CVCC, CCVCC.

3.2.1.1 Single level

The results in this section show that the use of the current set of operations in a Harmonic Serialism grammar results in a syllable structure typology that is quite similar to those predicted by parallel OT grammars. This step is necessary to confirm that the set of constraints and operations being used by the model will yield reasonable results for an existing framework of Harmonic Serialism, before proceeding to testing the novel properties of Stratal Harmonic Serialism. Harmonic Serialism makes different predictions than parallel OT in many cases, so it is not trivial to ensure that it does not make different predictions with Gradual Syllabification at a single level.

To find the predictions of the model at a single level, I ran a simulation finding the outputs of the possible grammars using six constraints: ParseSeg, NoCoda, Onset, Complex, Max, and IdSyl. For a single level of the grammar, there are *n!* unique total rankings of the constraints. With six constraints, this yields 720 different constraint rankings to be considered. To make the evaluation task more manageable, rankings were consolidated in two ways. First, all rankings with identical input-output mapping were consolidated, yielding 15 unique mappings. In this simulation, the input-output mapping pairs are based on a set of nine basic syllable types, with the the following set of unparsed input strings: CV, CVC, V, VC, CCV, VCC, CCVC, CVCC, CCVCC. Second, all mappings were consolidated into a set of syllable type inventories based on parsed syllables in the output, yielding 8 unique inventories, shown in Table 1.

No.	Syllable Types
1	cv
2	cv, ccv
3	cv, v
4	cv, cvc
5	cv, ccv, v
6	cv, cvc, v, vc
7	cv, cvc, ccv, ccvc, cvcc, ccvcc
8	cv, cvc, ccv, ccvc, cvcc, ccvcc, v, vc, vcc

Table 1: Inventories from a single level of the grammar with basic inputs

The difference in number of unique mappings is due to the different types of objects considered. For example in the case of rankings yielding the inventory containing only CV syllables, the first row in Table 1, there are two unique mappings. The first mapping is a result of a ranking like ParseSeg>>NoCoda>>Onset>>*Complex>>Max>>IdSyl, in which the low ranking of Max results in all extraneous unparsed segments being deleted after core syllabification. The second mapping is the result of a ranking like

NoCoda>>Onset>>*Complex>>Max>>ParseSeg>>IdSyl, in which the ranking of Max>>ParseSeg caused extraneous segments to remain unparsed after core syllabification. The differences between these two mappings are shown in Table 2.

Input	Output 1	Output 2
CV	(cv)	(cv)
CVC	(cv)	(cv)C
V	—	V
VC	—	VC
CCV	(cv)	C(cv)
VCC	—	VCC
CCVC	(cv)	C(cv)C
CVCC	(cv)	(cv)CC
CCVCC	(cv)	C(cv)CC

Table 2: Different mappings yielding the same inventory

While the outputs in these mappings are different, they both contain the same type of parsed syllable, namely (cv), in their inventories. For the outputs in the Output 1 column, all segments that cannot be parsed into (cv) syllables are deleted. In the Output 2 column, segments that cannot be parsed into (cv) syllables remain unparsed. The remaining unparsed segments in the Output 2 column are not included in the inventory because they are not associated with any syllable and will be removed by stray erasure. While Dep is not considered in this analysis, if it were, there would be a third column with output mappings, where segment insertion was used to create (cv) syllables from extraneous segments, with mappings such as V→ (cv) and CVC→ (cv)(cv). This prediction of different mappings resulting in the same inventories is not unique to Gradual

Syllabification or Harmonic Serialism. The same types of predictions would be made in a parallel OT analysis of syllabification using Max and Dep.

The set of predicted syllable type inventories in Table 1 is a good result because it captures all desired inventories and does not include any inventories that violate implicational laws, which I will use as a metric for evaluating syllable inventories due its acceptance as a description of empirical generalizations of possible syllable inventories in natural language. For example, in one typology of parametric variation in syllable types, all cases conform to these implicational laws (Blevins 1995). This notion of implicational laws was first proposed by Jakobson (1963) and extended in later works (Greenberg 1966; Clements and Keyser 1983). These theories of implicational laws were thought to best describe the empirical generalizations of syllable types, and exceptions to these generalizations are difficult to find. Implicational markedness was later formalized in Optimality Theory, with constraint interaction predicting the possible set of outputs (Prince and Smolensky 1993). Jarosz (2010) and Levelt, Schiller, and Levelt (2000) provide predictions of implicational markedness for syllable type inventories in acquisition. Here, I adopt a similar prediction of implicational markedness expected for the syllable type inventories, with a slight modification due to a different constraint set. In this typology, I use a simplified *Complex constraint, rather than separate constraints for complex onsets and codas, as used by Jarosz (2010), squashing some of the relations. The resulting implicational markedness interactions using the current constraint set are shown in Figure 7.

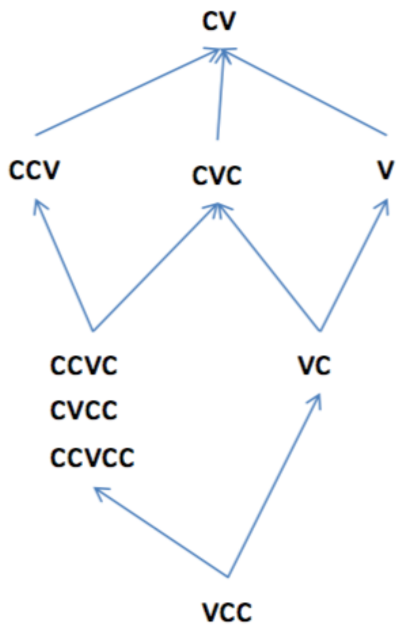

Figure 7: Implicational markedness of syllable types

The diagram in Figure 7 shows the implicational relationship of syllable types based on markedness. The (cv) syllable is at the top because it is implied by all syllable types, and predicted to appear in all syllable inventories. For example, the arrow between (ccv) and (cv) indicates that any inventory containing (ccv) will also contain (cv). An inventory like *{ccv} violates implicational markedness and is predicted not to exist by implicational laws or parallel OT, nor is it an attested as a possible syllable inventory in natural language. This implication is unidirectional; while (ccv) implies (cv), (cv) does not imply (ccv), thus the existence of the inventory {cv}. Syllable types that are not connected, such as (ccv) and (cvc), do not have an implicational relationship, so they can occur in without one another in inventories like {cv, ccv} and {cv, cvc}. Using these implicational rules, certain syllable type inventories are predicted not to exist.

Using implicational rules and the marked structures present in the analysis, we can construct a lattice to enumerate all the syllable type inventories that we want to be generated by the Gradual Syllabification model. There are three marked structures, violating NoCoda, Onset, and *Complex, which result in 2^3 points on the lattice, shown in Figure 8.

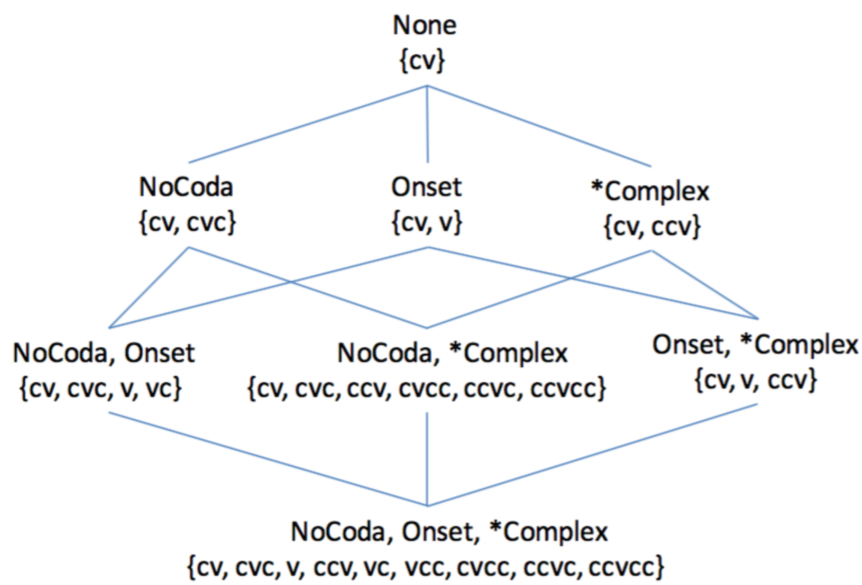

Figure 8: Syllable type inventories

At each point in the lattice, the is a combination of marked structures, indicated by the markedness constraints they violate. The point on the top row violates no markedness constraints, while the point on the bottom violates all three markedness constraints. In addition to marked structures, each point also contains the syllable inventory containing all syllables which maximally violate the listed constraints. For example, the first point in the third row contains the marked structures of codas and onsetless syllables, resulting in the inventory {cv, cvc, v, vc}. We could imagine a possible inventory *{cv, vc} that also contains the marked structures of codas and onsetless syllables, but this inventory violates implicational markedness because it lacks the syllables (v) and (cvc) implied by (vc).

The eight syllable type inventories in Figure 8 predicted by the laws of implicational markedness are the same as the eight inventories in Table 1, which are the inventories generated by Gradual Syllabification in a single level of Stratal Harmonic Serialism. These results confirm that the theory of Gradual Syllabification is sufficiently generative and restrictive; the model produces all the eight inventories expected from the lattice structure and does not produce any additional inventories that violate implicational markedness. In the next section, I extend these results to test the property of interaction between multiple levels in the model.

3.1.1.1 Multiple levels

When two levels of the grammar are used, the model is able to arrive at a set of inventories identical to that found after a single level of the grammar. In this simulation, the outputs from the first level of the grammar are passed through to another level of a Harmonic Serialism grammar as inputs, with a reranking of the constraints. The major difference between the first and second levels is that the inputs to the first level were not parsed, whereas at the second level the inputs have preexisting structure from parsing at the first level. The syllable inventories after two levels of the grammar are shown in Table 3.

No.	Syllable Types
1	cv
2	cv, ccv
3	cv, v
4	cv, cvc
5	cv, ccv, v
6	cv, cvc, v, vc
7	cv, cvc, ccv, ccvc, cvcc, ccvcc
8	cv, cvc, ccv, ccvc, cvcc, ccvcc, v, vc, vcc

Table 3: Inventories from two levels of the grammar

These results are identical to those found at a single level of the grammar, with all the same benefits discussed in the previous section. Given the marked structures used in the analysis, there are no expected inventories that are missing, as predicted by the lattice shown in the previous section. Additionally, there are no inventories that violate implicational markedness. Thus there is no overgeneration or undergeneration of inventories with two levels of the grammar.

The results shown here are very encouraging, but also a bit surprising. How is it possible that the ranking of any input can be changed and still come out with the same set of results? This result follows from a property of structure preservation, neutralization at the final level, and a limited set of inputs at this level. To understand how these properties generate the inventories seen here, we can consider a number of possible combinations of constraint rankings, and demonstrate that they do not generate any additional inventories beyond those found at a single level of the grammar.

The simplest case is one in which the ranking at the first level is identical to that of the second level, or with the same crucial rankings such that they would generate the same inventory. In this case, it is easy to see that no additional inventories would be generated because a change in syllabification between the levels would not be harmonically improving.

Another possibility in ranking combination is one in which an earlier level permits a marked structure that is not permitted by a later level. Consider a grammar where the initial level returns the inventory {cv, cvc} and the second level does not permit codas, such as by the ranking of NoCoda>>Max, resulting in a final inventory of {cv}.

(173) Inventory neutralization: deletion of codas

/(cvc)/	ParseSeg	NoCoda	Onset	*Complex	Max	IdSyl
a. (cvc)		W1			L	
b. →(cv)					1	

The tableau in (173) demonstrates that marked structures from earlier levels can be deleted in later ones, in this case the deletion of codas. In the case of this ranking, the (cvc) syllables in the input neutralize to (cv) syllables. This neutralization changes the inventory, but does not introduce any new inventory type to the typology.

Another property that comes into play is structure preservation. In such a case, the second level has a ranking that would predict a different syllable structure at the initial level, but cannot effect any change to already parsed syllables, due to an absence of any unparsing operations in the set of possible operations. Assume we still have an initial level with an inventory of {cv,cvc}, consider the ranking of Max>>NoCoda>>ParseSeg in tableau (174).

(174) Structure preservation: no deletion of codas

/(cvc)/	Max	NoCoda	Onset	*Complex	ParseSeg	IdSyl
a. →(cvc)		1				
b. (cv)	W1	L				

In this case, the ranking of Max>>NoCoda prevents deletion of existing codas in the input. What is interesting about this case is that the ranking would not allow codas under different circumstances. Consider the following Level 1 derivation, under the same ranking:

(175) Step 1: Core syllabification

/CVC/	Max	NoCoda	Onset	*Complex	ParseSeg	IdSyl
a. CVC					W3	
b. →(cv)C					1	

(176) Step 2: Convergence

/(cv)C/	Max	NoCoda	Onset	*Complex	ParseSeg	IdSyl
a. →(cv)C					1	
b. (cvc)		W1			L	

In the first step of the derivation, the winner is the candidate undergoing core syllabification. Note that at the first level, the input is the unparsed /CVC/, rather the parsed /(cvc)/ at the second level, so we have a different set of possible operations at this first step. In the second step, the possibility to adjoin the coda in candidate (b) is not harmonically improving, due to the ranking NoCoda>>ParseSeg. The winner at step 2 is the faithful candidate (a), resulting in convergence. The final output here is a (cv) syllable, rather than a (cvc) one that we find with the same ranking at the second level. This result falls under the property of structure preservation, wherein parsed syllables cannot be undone even though they would be prohibited by the same ranking under different circumstances, crucially due to the lack of an unparsing operation[14]. Thus the resulting inventory of {cv,cvc} remains unchanged, and no new inventory types have been added to the typology.

A final property of the inventory at the second level is the lack of sufficient inputs, which were reduced as outputs of the first level. Once again let us consider an initial inventory of {cv,cvc} at the second level. Consider a ranking where ParseSeg and Max dominate the set of markedness constraints {NoCoda, Onset, *Complex}, or M. This ranking would permit the full range of syllable types at the first level of the grammar, due to the pressure to parse marked syllables from ParseSeg>>M, and the inability to delete them, from Max>>M. However, this permissive ranking does not do anything to alter the existing syllables at the second level because there is no motivation to do so.

[14] Simulations with an unparsing operation will be discussed in section 4.5.

(177) Insufficient inputs: no new syllables

/(cvc)/	ParseSeg	Max	NoCoda	Onset	*Complex	IdSyl
a. →(cvc)			1			
b. (ccvc)			1		W1	
c. (vc)		W1	1	W1		

As one would expect, there is no reason for the grammar to create new marked structures, and candidates which do so are harmonically bounded by the faithful candidate. Thus with a more permissive ranking, there are no new inventory types added to the typology.

From the properties described here, I have demonstrated why it is the case that no new inventory types are generated at the second level of the grammar when no affixal material is added. At two levels of the grammar, the model is able to produce a set of inventories that is well-restrained. Like the results at a single level of the grammar, it does not undergenerate by missing crucial inventories that should be included, nor does it overgenerate by producing inventories that violate implicational markedness.

3.1.1.1 Affixes

The next important simulation is the result of the model with affixes added in at the second level of the grammar. The set of affixes used is {C, CC, CV, CVC, CCV, V, VC, VCC, CCVC, CVCC, CCVCC}, the nine basic syllable types plus consonantal affixes{C, CC}. While showing that multiple levels of the grammar with constraint reranking is important to demonstrate the restrictiveness of one powerful component of the grammar, the introduction of new unparsed affixal material is also a powerful component. In these simulations, the model produces a larger set of inventories than seen in the previous two levels, with 17 total inventories, shown in Table 4.

No.	Syllable Types
1	cv
2	cv, ccv
3	cv, v
4	cv, cvc
5	cv, ccv, v
6	cv, cvc, v, vc
7	cv, cvc, ccv, ccvc, cvcc, ccvcc
8	cv, cvc, ccv, ccvc, cvcc, ccvcc, v, vc, vcc
9	cv, cvc, v
10	cv, cvc, ccv
11	cv, cvc, ccv, v
12	cv, cvc. ccv, ccvc
13	cv, cvc, ccv, v, vc
14	cv, cvc, ccv, ccvc, v
15	cv, cvc, ccv, ccvc, v, vc
16	cv, cvc, ccv, ccvc, cvcc, ccvcc, v
17	cv, cvc, ccv, ccvc, cvcc, ccvcc, v, vc

Table 4: Inventories from two levels of the grammar with affixes and basic inputs

These results show that the affixal component of the model is less restrictive than the effect of multiple levels, but in assessing these additional inventories, I will show that they are valid inventories and not problematic for the theory.

Out of 17 total inventories, the first 8 are identical to those generated by the previously discussed components of the model. These inventories are predicted as would be expected of the model, so there is no regression to undergeneration of inventories. The 9 new inventories will need to be accounted for. From an Optimality Theory perspective, parallel or serial, these results seem unexpected. Consider an inventory like #9, {cv, cvc, v}. Under a ranking where violations of NoCoda and Onset are both permitted in a syllable, it seems to follow that (vc) will be permitted as well, illustrated in (178).

(178) Non-additive markedness

		Faith	NoCoda	Onset
/CV/	→(cv)			
	(v)	*!		*
/CVC/	→(cvc)		*	
	(cv)	*!		
/V/	→(v)			*
	(cv)	*!		
?/VC/	→(vc)		*	*
	(v)	*!		*

The ranking of Faith>>NoCoda and Faith>>Onset should imply that (vc) is a member of this inventory, while here it is not. While this is an important theoretical assumption, and one that has held at a single level of the grammar, the omission of (vc) from the inventory {cv, cvc, v} is not a violation of implicational markedness. Implicationality only operates in one direction, as shown in Figure 8. While (vc) implies (v) and (cvc), (cvc) and (v) do not imply (vc).

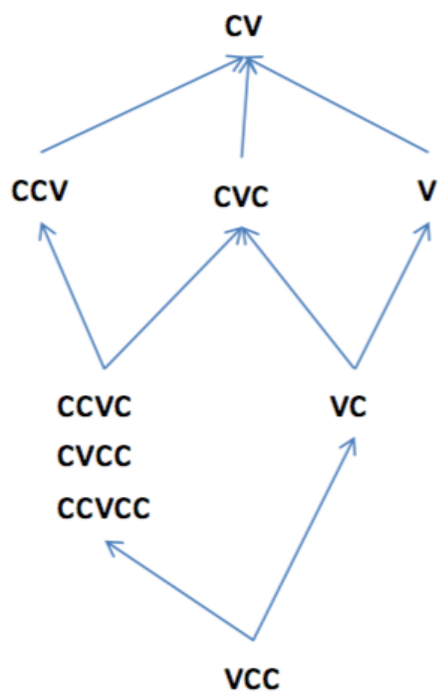

Figure 4: Implicational Markedness

An inventory that would violate implicational markedness would be {cv, cvc, vc}, which contains the syllable (vc), but crucially does not contain the syllable (v). Thus far we have used the measure of implicational markedness to judge the validity of a given inventory. Though the additional 9 inventories seem unexpected, none of them violate implicational markedness.

The path to these new inventories occurs when adding new unparsed material to a set of existing syllabified structures, and having a different ranking of constraints govern the new pattern of syllabification. Let us examine this for one case, the inventory {cv, cvc, ccv, ccvc} and one possible path for how it was derived, particularly with regard to how we can derive a syllable like (ccvc), but not the expected (cvcc).

Level	Ranking	Input	Output
1	ParseSeg>>NoCoda>>Onset>>Max>>*Complex>>IdSy	All inputs	cv, ccv
2	Onset>>Max>>*Complex>>ParseSeg>>NoCoda>>IdSyl	{cv, ccv}+affixes	cv, ccv, cvc, ccvc

Table 3: Derivation of {cv, cvc, ccv, ccvc} inventory

The first stage of the derivation is a pass of the set of all inputs, the nine basic syllable types {CV, CVC, CCV, V, VC, VCC, CCVC, CVCC, CCVCC}, through the first level of the grammar. In this case, we have the ranking ParseSeg>>NoCoda>>Onset>>Max>>*Complex>>IdSy, resulting in the intermediate outputs {cv, ccv}. In the second stage, the intermediate outputs {cv, ccv} are combined with the set of affixes {C, CC, CV, CVC, CCV, V, VC, VCC, CCVC, CVCC, CCVCC}. These inputs are passed through a second level of the grammar with the ranking Onset>>Max>>*Complex>>ParseSeg>>NoCoda>>IdSyl, resulting in the final output inventory {cv, ccv, cvc, ccvc}.

At the first level, the only marked structure permitted is complex clusters. The ranking of ParseSeg>>Max>>*Complex permits complex clusters to be parsed, rather than be deleted or left unparsed, while ParseSeg>>NoCoda>>Onset>>Max does not allow codas or onsetless syllables because they are deleted. At the second level, the permissible marked structures change. Under the ranking *Complex>>ParseSeg, the parsing of new complex clusters is not harmonically improving, so they cannot be parsed. Existing clusters cannot be deleted, due to Max>>*Complex. Additionally, codas are now permitted, due to the ranking Max>>ParseSeg>>NoCoda. With only {cv, ccv} as inputs, this new ranking would not result in any change to the inventory. The existing clusters cannot be changed, and there is no reason to spontaneously generate codas. However, with the addition of affixes at this level, we do see a change in the inventory. In addition to the inputs /(cv)/ and /(ccv)/, there are also inputs like /(cv)-C/ and /(ccv)-C/, which are parsed into (cvc) and (ccvc) syllables. We also consider inputs like /CV-cc/, but this affix cannot be parsed into a complex cluster, due to the ranking *Complex>>ParseSeg, thus explaining the absence of (cvcc) in the inventory.

To summarize the interactions resulting in the inventory {cv, ccv, cvc, ccvc}, this is the result of two levels of marked structures combining. Complex onsets are created at the first level, but no simple or complex codas are created at this level due to the high ranking of NoCoda. Simple codas are created only at the second level, or in other words, codas are always the result of affixal material, and only predicted to occur at affix boundaries. Complex codas are not parsed at any level, because of the high ranking of NoCoda at the first level and the high ranking of complex at the second level, so they are not present in the inventory. This results in an inventory that is different from what we expect from single level Optimality Theory grammars, but is not in violation of implicational markedness.

3.2.2 Rich inputs

In this section, I provide results of a simulation with a richer set of inputs. In addition to the nine basic syllable type inputs used in the previous section (CV, CVC, V, VC, CCV, VCC, CCVC, CVCC, CCVCC), I also include all pairwise combinations of these strings, resulting in 90 total inputs.[15] The resulting simulation has some different results, with two

[15] The complete set of rich inputs is as follows: CV, CVC, V, VC, CCV, VCC, CCVC, CVCC, CCVCC, CVCV, CVCVC, CVV, CVVC, CVCCV, CVVCC, CVCCVC, CVCVCC, CVCCVCC, CVCCV, CVCCVC, CVCV, CVCVC, CVCCCV, CVCVCC, CVCCCVC, CVCCVCC, CVCCCVCC, VCV, VCVC, VV, VVC, VCCV, VVCC, VCCVC, VCVCC, VCCVCC, VCCV, VCCVC, VCV, VCVC,

additional inventories as an effect of levels and three fewer inventories as an effect of levels with affixes.

3.2.2.1 Single level

At a single level of the grammar, the resulting set of syllable inventories from rich inputs is the same as with basic inputs, shown in Table 4:

No.	Syllable Types
1	cv
2	cv, ccv
3	cv, v
4	cv, cvc
5	cv, ccv, v
6	cv, cvc, v, vc
7	cv, cvc, ccv, ccvc, cvcc, ccvcc
8	cv, cvc, ccv, ccvc, cvcc, ccvcc, v, vc, vcc

Table 4: Inventories from a single level of the grammar with rich inputs

As with the previous results, these inventories are a desired result to due their conformity to implicational markedness.

VCCCV, VCVCC, VCCCVC, VCCVCC, VCCCVCC, CCVCV, CCVCVC, CCVV, CCVVC, CCVCCV, CCVVCC, CCVCCVC, CCVCVCC, CCVCCVCC, VCCCV, VCCCVC, VCCV, VCCVC, VCCCCV, VCCVCC, VCCCCVC, VCCCVCC, VCCCCVCC, CCVCCV, CCVCCVC, CCVCV, CCVCVC, CCVCCCV, CCVCVCC, CCVCCCVC, CCVCCVCC, CCVCCCVCC, CVCCCV, CVCCCVC, CVCCV, CVCCVC, CVCCCCV, CVCCVCC, CVCCCCVC, CVCCCVCC, CVCCCCVCC, CCVCCCV, CCVCCCVC, CCVCCV, CCVCCVC, CCVCCCCV, CCVCCVCC, CCVCCCCVC, CCVCCCVCC, CCVCCCCVCC

3.2.2.2 Multiple levels

The results with rich inputs at multiple levels do show a level effect, with two additional inventories predicted [cv, cvc, ccv, ccvc] and [cv, cvc, ccv, ccvc, v, vc], as shown in Table 5.

No.	Syllable Types
1	cv
2	cv, ccv
3	cv, v
4	cv, cvc
5	cv, ccv, v
6	cv, cvc, v, vc
7	cv, cvc, ccv, ccvc, cvcc, ccvcc
8	cv, cvc, ccv, ccvc, cvcc, ccvcc, v, vc, vcc
9	cv, cvc, ccv, ccvc
10	cv, cvc, ccv, ccvc, v, vc

Table 5: Inventories from multiple levels of the grammar with rich inputs

These two additional inventories are not completely new, as they both appeared in the results of the results of multiple levels with affixes produced with basic inputs, and they are derived in much the same way, through additive markedness. Both these inventories share a common trait: each has codas and complex onsets, but not complex codas. Consider one case, in which the output from level 1 contains codas, but not complex segments, so the set of inputs to level 2 are [(cv), (cvc), (cv)(cv), (cvc)(cvc), (cv)(cvc), (cvc)(cv)]. Now also consider that a reranking of constraints at level 2 that permits complex clusters, but not codas. Additionally, this ranking has permits resyllabification as a repair for now dispreferred codas, but not deletion. One such ranking is ParseSeg>>Max>>NoCoda>>Onset>>*Complex>>IdSyl, as shown in tableau (179), which shows how new complex onsets would be formed through resyllabification.

(179) Step 1: Complex onsets through resyllabification

/(cvc)(cvc)/	ParseSeg	Max	NoCoda	Onset	*Complex	IdSyl
a. (cvc)(cvc)			W2		L	L
b. →(cv)(ccvc)			1		1	1
c. (cvcc)(vc)			W3	W1	1	1
d. (cv)(cvc)		W1			L	L

In this derivation, the winning candidate (b) loses a violation of NoCoda at the expense of lower ranked *Complex and IdSyl. Note that the reverse resyllabification case, as shown in candidate (c) would not be possible under any ranking, as it has a gratuitous violation of Onset, and would always be harmonically bounded by the faithful candidate. In the next step, the derivation converges, unable to delete the remaining coda.

(180) Step 2: Converge

/(cv)(ccvc)/	ParseSeg	Max	NoCoda	Onset	*Complex	IdSyl
a. →(cv)(ccvc)			1		1	
b. (cv)(ccv)		W1			1	

This derivation shows how new syllable inventories are formed as an effect of levels when the inputs are rich enough to show resyllabification effects between syllable boundaries.

3.2.2.3 Multiple levels with affixes

In this section, I provide results for multiple levels with affixes with rich inputs. As in the previous section, these results are slightly different from those found with basic inputs, but here there are fewer syllable type inventories, 14 rather than 17.

No.	Syllable Types
1	cv
2	cv, ccv
3	cv, v
4	cv, cvc
5	cv, ccv, v
6	cv, cvc, v, vc
7	cv, cvc, ccv, ccvc, cvcc, ccvcc
8	cv, cvc, ccv, ccvc, cvcc, ccvcc, v, vc, vcc
9	cv, cvc, v
10	cv, cvc. ccv, ccvc
11	cv, cvc, ccv, ccvc, v
12	cv, cvc, ccv, ccvc, v, vc
13	cv, cvc, ccv, ccvc, cvcc, ccvcc, v
14	cv, cvc, ccv, ccvc, cvcc, ccvcc, v, vc

Table 6: Inventories from two levels of the grammar with affixes and rich inputs

The three inventories not predicted with rich inputs, as opposed to basic inputs, are [cv, cvc, ccv], [cv, cvc, ccv, v], and [cv, cvc, ccv, vc]. Each of these three cases has a corresponding inventory with an additional ‘ccvc’ syllable, the same syllable type added in the previous level. It is not surprising that the addition of that syllable type would correspond to the loss of these three undesired inventories, so this prediction is a net benefit.

3.3 Theoretical implications

In the previous sections, I have presented the results of the model when using the defined set of constraints and operations. The specific formulation of these constraints and operations is necessary because other variations produce undesirable results. In this section, I outline some simulations in which undesirable results were achieved. These results demonstrate the necessity of specific formulations of constraints and operations, with small changes to these definitions having significant consequences. Additionally, this work brings up the importance of having complete typologies, which may produce interactions that are unexpected (Karttunen 2006; Bane and Riggle 2012). In this section, I discuss three such interactions, involving the NoCoda constraint, the deletion operation, and unparsing operations.

A common pattern among these cases is the prediction of pathological locally-optimal interactions. The tendency of serial OT to predict locally-optimal grammars, as compared to globally-optimal ones predicted by parallel OT, has been shown to be beneficial in many domains including cluster reduction (McCarthy 2008), harmony (McCarthy 2009), and stress (Pruitt 2010, 2012).

An example of the locality problem is demonstrated by sour-grapes spreading predicted for nasal harmony, as in Jahore Malay (McCarthy 2009).

(181) Nasal harmony in Jahore Malay (Onn 1976) via McCarthy (2009)

a. mã̃ʔ̃ãp 'pardon'

b. mə̃ratappi 'to cause to cry'

In (181), the nasal feature spreads rightward from the initial nasal *m* until it hits the segment *p*, which cannot be nasalized. All vowels in this case can be nasalized because *ʔ* is not a blocker. In (181), the nasal feature can only spread rightward to one segment, *ə*, because the following segment, *r*, is a blocker. The remaining vowels in the word are not nasalized. As this nasal harmony is attested in Jahore Malay, both parallel and serial OT should be able to account for it. However, McCarthy (2009) shows that this pattern cannot be accounted for in parallel OT, instead predicting sour-grapes spreading, where nasal harmony only applies if it can spread across the entire word, making the form in (181) *məratappi* instead of the desired *mə̃ratappi*. This type of sour-grapes spreading is not attested, and thus an undesired prediction. The difference between the parallel and serial OT predictions are due to locality. Because parallel OT considers all possible candidates at once, it selects the globally-optimal candidate, in this case the candidate that minimizes violations of the Agree and Id-Nasal constraints. On the other hand, serial OT considers only minimally-different candidates at each step, which may not lead to the globally-optimal candidate, instead arriving at the locally-optimal one. In this case, if nasal spreading is optimal in the first step, the path to sour-grapes candidate becomes blocked. While the locality property is beneficial in the nasal spreading case, it is potentially problematic in the syllable structure domain, as I will demonstrate in this section.

3.3.1 Formulation of NoCoda

One of the assumptions made in the simulation reported above is a slight change in the assignment of violations by the NoCoda constraint. In this simulation, NoCoda assigns a violation to each instance of a coda consonant, rather than assigning a violation to each syllable containing a coda. The crucial difference is in cases of complex codas like (cvcc), where the current version of NoCoda would assign two violations, but the traditional interpretation of NoCoda would assign only one violation. This formulation is used to prevent a typological pattern that violates implicational markedness.

In this case, the violation of implicational markedness occurs in inventories that contain complex codas, but no simple codas. This is due to rankings which produce mappings like the following:

(182) Mappings leading to unattested inventory

/(cvc)/ → (cv)

/(cvcc)/ → (cvcc)

For the input /(cvc)/, the coda is deleted resulting in the output (cv), while the input /(cvcc)/ does not undergo deletion. Note that the example in (182) contains parsed syllables as inputs because this mapping only arises when generating typologies for two or more levels of the grammar. The traditional interpretation of NoCoda is not a problem for parallel of serial versions of Optimality Theory at a single level.

Let us consider one possible ranking that will produce the mapping in (182) when using the traditional interpretation of NoCoda.

(183) No Simple Codas with NoCoda(old)

/(cvc)/	NoCoda(old)	Max	*Complex	ParseSeg	Onset	IdSyl
a. (cvc)	W1	L				
b. →(cv)		1				

(184) Complex codas with NoCoda(old)

/(cvcc)/	NoCoda(old)	Max	*Complex	ParseSeg	Onset	IdSyl
a. →(cvcc)	1		1			
b. (cvc)	1	W1	L			

In tableau (183), the ranking NoCoda(old)>>Max permits deletion of a coda consonant. In tableau (184), the ranking Max>>*Complex does not permit deletion to repair the complex coda. The deletion of a coda consonant in candidate (b) does not remove any violations of NoCoda(old).

This mapping is a result of the gradualness property of Harmonic Serialism and assumptions about only deleting a single segment at once. While the ranking NoCoda>>Max tells us that deletion of coda consonants should be preferred, the intermediate step of reducing the complex coda to a singleton coda is not harmonically improving.

The resulting inventory is problematic because these types of inventories do not follow implicational markedness and are unattested. There are no languages which permit complex codas, but not simple codas. Our theory would require a solution to this problem. The approach taken here is a slight change to the traditional interpretation of violation assignments of the NoCoda constraint. Rather than assigning a single violation of NoCoda to a syllable containing a coda, this version of NoCoda assigns a violation for each occurrence of a segment in a coda.

Now consider this ranking with the new interpretation of NoCoda, in tableaux (185) and (186).

(185) No Simple Codas with NoCoda(new)

/(cvc)/	NoCoda(new)	Max	*Complex	ParseSeg	Onset	IdSyl
a. (cvc)	W1	L				
b. →(cv)		1				

(186) No Complex codas with NoCoda(new)

/(cvcc)/	NoCoda(new)	Max	*Complex	ParseSeg	Onset	IdSyl
a. (cvcc)	W2	L	1			
b. →(cvc)	1	1				

The tableau in (185) remains the same as with the previous interpretation, but tableau (186) has a different outcome. With the complex coda in candidate (a) receiving two violations of NoCoda, the deletion of one of the coda consonants in candidate (b) is harmonically improving. With this step of deletion to singleton coda, the next step in the derivation will be deletion of the remaining coda consonant, followed by convergence. This assignment of violations allows the gradual deletion of a coda to be harmonically improving under NoCoda>>Max, regardless of the ranking of *Complex, thus eliminating the undesired mapping seen in (182).

The solution of slightly modifying the definition of NoCoda does not introduce any new problems into the syllable type typology. The results presented earlier show the complete typology, which does not violate implicational markedness in any way, using this modified definition discussed here. Of course, this is not the only possible solution and there are alternatives that one might consider. One possible solution is allowing the deletion of both consonants simultaneously as a single operation. This would remove the intermediate step from blocking the second step. A simulation run with this operation removes spurious inventories. Another possible solution would be a reworking of the constraint *Complex. Currently a single constraint penalizes both complex onsets and complex codas. In other accounts, this constraint may be split into two separate constraints, *ComplexOnset and *ComplexCoda. If *ComplexCoda is a more specific variant of NoCoda, we can impose a meta-ranking of *ComplexCoda>>NoCoda, which would prevent the mappings in (182), caused by NoCoda>>Max>>*Complex. While these solutions are both plausible, the current solution of adjusting NoCoda violations is judged to be optimal because it removes the undesirable mapping without introducing any additional constraints or operations.

The proposal to modify NoCoda is able to remove the undesired mapping from the typology without any negative effects. It is worth noting the implications of this mapping on the theory. Were we unable to solve this problem, the prediction of a mapping in violation of implicational markedness would likely be fatal to the framework. While there is a solution for this particular case, it is important to take note of the type of problem and the potential for a benefit of a theory to also be a flaw. In this case, under a particular set of circumstances we are able to create a mapping wherein (cvc) syllables become (cv) syllables, while (cvcc) syllables stay as they are. This pattern is a locally optimal one. While a globally optimal grammar would need to continue deletion of the complex coda, or be stuck in a sour-grapes situation with both singleton and complex codas, this grammar is stuck in the middle, deleting singleton codas but retaining complex ones. The property of permitting locally optimal grammars and excluding globally optimal ones is an important benefit of Harmonic Serialism in many domains, but this property also has the potential to make bad predictions. This does not cause any immediate problems for the current proposal, but is something to be aware of when considering the typological implications of a set of constraints and operations in Harmonic Serialism and related frameworks.

3.3.2 Deletion of V syllables

In the previous section, I presented a case in which an inventory in violation of implicational markedness could be generated depending on the formulation of constraints banning complex codas. Here I will present a similarly problematic case dealing with the structure of onsetless syllables, leading to inventories like {cv, cvc, vc}. This inventory is problematic because it contains a (vc) syllable, but no (v) syllable. Inventories of this type can occur due to mappings with deletion of (v) syllables, but no deletion in (vc) syllables, as shown in (187):

(187) Mappings leading to unattested inventory

/(v)/ → Ø

/(vc)/→ (vc)

These mappings occur when onsetless syllables are dispreferred over deletion, as due to a ranking of Onset>>Max, but an intermediate step of coda deletion is not harmonically improving, due to a ranking of Max>>NoCoda. As in the case of the deletion of complex codas, the inputs in this mapping are parsed syllables. This problem is novel because these particular pathologies do not occur in single level grammars, though it is possible that similar cases could occur in a standard Harmonic Serialism grammar.

Let us consider one such ranking and how it produces this mapping. The tableau in (188) shows the first mapping in which a single onsetless vowel deletes, while the tableau in (189) shows a failure to delete an onsetless vowel when it is followed by a coda.

(188) No onsetless syllables

/(v)/	Onset	Max	NoCoda	*Complex	ParseSeg	IdSyl
a. (v)	W1	L				
b. →Ø		1				

(189) Onsetless syllables with codas

/(vc)/	Onset	Max	NoCoda	*Complex	ParseSeg	IdSyl
a. →(vc)	1		1			
b. (v)	1	W1	L			

From the ranking Onset>>Max, the grammar seems as though it would prefer to remove violations of Onset through deletion, as it does in the mapping /(v)/→ Ø. In the case of the input /(vc)/, the step of vowel deletion is blocked by the presence of a coda.

This case bears a resemblance to the previous example of coda deletion, in which a locally optimal inventory is generated due to the blocking of an intermediate form that causes a derivation to converge prematurely. This case is more complex because rather than modifying a constraint, it requires making important assumptions about the structure of the syllable types and what operations are permitted. Even in this example, we are making the crucial assumption that (vc)→(v) is a possible operation, but (vc)→(c) is not.

To find a solution to this problem, the structure of these syllable types will need to be more thoroughly fleshed out. There are two possible paths to a solution here: permit deletion of the v in (vc) to reach a globally optimal inventory or prohibit deletion of the v in (v) as a kind of sour-grapes approach.

First, let us consider the case in which we permit deletion of the v in (vc). This would likely involve a mapping of the sort (vc)→(c). The problem here is the remaining structure, (c). What is it? There are at least two possibilities for this structure. If it has a mora it could be thought of as a syllabic consonant, but that would still violate Onset and this mapping would not be harmonically improving. If it does not have a mora, we could consider this to be a minor syllable. There are at least two problems with this approach. First, introducing minor syllables would require a constraint militating against them; let's call this constraint *Minor. Under some rankings, we would be able to delete V to produce a syllable like (v), as in tableau (190).

(190) Deletion into a minor syllable

/(vc)/	Onset	Max	*Minor	NoCoda	*Complex	ParseSeg	IdSyl
a. (vc)	W1	L	L	1			
b. (v)	W1	1	L				
c. →(c)		1	1				

In this case the ranking Onset>>*Minor permits the removal of an Onset violation at the expense of creating a marked minor syllable. However, the factorial typology will also include rankings with *Minor>>Onset, as shown in (191).

(191) Failure to delete into a minor syllable

/(vc)/	*Minor	Onset	Max	NoCoda	*Complex	ParseSeg	IdSyl
a. →(vc)		1		1			
b. (v)		1	W1	L			
c. (c)	W1	L	W1	L			

In this case, the ranking does not permit the removal of an Onset violation at the expense of incurring a new *Minor violation. Under this ranking, we will still have the problematic mappings of /(v)/→Ø and /(vc)/→(vc), even with the addition of the minor syllable. While not completely ruling out the possibility for further work on this question, the possibility of a (vc)→(c) mapping does not provide a straightforward solution to the inventory problem.

Let us return to the second possibility, blocking the possibility of the /(v)/→Ø mapping, which is the approach taken here. The intuition behind this blocking is that the deletion of an entire syllable is not a gradual step, and thus cannot be a single operation. Instead, the path toward the deletion of the vowel must occur in multiple harmonically-improving operations, and crucially, the first of these operations does not repair the violation of Onset. This could be the deletion of a mora, the deletion of a place feature, or delinking from either of these. For now, let us just assume there is some operation that deletes a feature from (v) and violates the constraint Faith.

(192) Blocking of (v)→Ø

/(v)/	Onset	Faith	Max	NoCoda	*Complex	ParseSeg	IdSyl
a. →(v)	1						
b. $(v)_{[-feature]}$	1	W1					

In tableau (192), we can see what the violation profile would be for such an operation. Regardless of the ranking of Faith, the loser $(v)_{[-feature]}$ is harmonically bounded by the winner (v). Unlike the *Minor constraint, when this Faith constraint is added the undesired mapping cannot be generated under any ranking. Using this approach of forcing the gradual deletion of (v) is able to solve the problem at hand.

This still leaves the remaining question of what exactly is this set of gradual operations that permit the deletion of (v). An important part of this approach is that Onset cannot be the markedness constraint responsible for compelling the deletion of (v). To be clear, this does not mean that deletion cannot be used to satisfy Onset in Onset>>Max rankings, which would be a disastrous failure of the theory. Deletion can be used freely under this ranking for all inputs under richness of the base, as long as the vowels are unparsed, as they are at the first level of the grammar. This operation only comes into play when the vowel in question is parsed. Furthermore, this does not mean that (v) is always prevented from being deleted. On the contrary, the ability to delete parsed vowels is crucial, particularly in the domain of stress. This theory does predict that additional markedness constraints other than Onset will be needed to compel the deletion of (v). Because stress systems are outside the scope of this chapter, I will not be exploring this question further and it will remain a topic for future research. For the purpose of the current study, the typology presented earlier in the chapter does make use of this theory of operations.

3.3.3 Unparsing

In the earlier results of section 3.2, I presented structure preservation as an important property of the model. This property crucially relied on an assumption that unparsing was not a possible operation. There is theoretical precedent for this question of structure building serial OT models. Work on prosodic structure in Harmonic Serialism has proposed allowing such structure to be built, but not unbuilt (Pruitt 2010). For the current work, this remains an empirical question on the types of inventories that are generated under either assumption.

In addition to the results discussed earlier, I ran a simulation testing the possibility of a model which does permit an unparsing operation. The unparsing operation permits a segment in a syllable to be disassociated with that syllable at the expense of one violation of ParseSeg. This operation is harmonically improving when it is dominated by a relevant markedness constraint, such as NoCoda, shown in tableau (193).

(193) Unparsing operation

/(cvc)/	NoCoda	ParseSeg
a. (cvc)	W1	L
b. →(cv)C		1

The unparsing model ran into some of the same problems, discussed in sections 3.3.1 and 3.3.2, namely locally optimal interactions resulting in the inventories containing {cvcc, *cvc} and {vc, *v}. In these cases, the unparsing operation caused the same type of problem as the deletion operation, resulting in inventories that violate implicational markedness.

The problematic inventory {cvcc, *cvc} is resolved by the same solution used to eliminate this inventory in section 3.3.1, the reformulation of the NoCoda constraint. In that case, the deletion operation permitted the mapping (cvc)→ (cv), but not the mapping *(cvcc)→(cvc)→(cv), which was blocked due to the lack of harmonic improvement for a NoCoda constraint with only a single violation of complex codas. The same situation occurs with the unparsing operation under the old definition of NoCoda, which permits the mapping (cvc)→(cv)C, but not *(cvcc)→(cvc)C→(cv)CC. With the new definition, the mapping (cvcc)→(cvc)C→(cv)CC is permitted, thus removing the problematic inventory {cvcc,*cvc} from the typology. This case is easily resolved and not fatal to the possibility of an unparsing operation in the model.

The inventory {vc,*v} presents a larger problem for the operation of unparsing. As in the case discussed in section 3.3.2, this inventory is the result of the mapping (v)→V, but not *(vc)→(v)→V, occurring. The tableaux in (194)-(195) show a possible ranking resulting in these mappings with the unparsing operation.

(194) Unparsing of (v)

/(v)/	Onset	ParseSeg	Complex	Max	NoCoda	IdSyl
a. (v)	W1	L				
b. →V		1				

(195) No unparsing in (vc)

/(vc)/	Onset	ParseSeg	Complex	Max	NoCoda	IdSyl
a. →(vc)	1				1	
b. (v)C	1	W1			L	
b. (v)	1			W1	L	

In (194), the ranking of Onset>>ParseSeg makes the unparsing of the onsetless V harmonically improving. In (195), ParseSeg>>NoCoda does not permit unparsing of the coda consonant, an important intermediate step for removing onsetless syllables. The coda cannot be removed through deletion either, due to the Max>>NoCoda ranking, so the derivation converges on (vc), resulting in the problematic inventory containing {vc, *v}.

The solution for this situation is not as straightforward as the NoCoda case. Recall that the same {vc, *v} inventory occurred with the deletion operation, and was resolved by preventing the deletion of an entire syllable in a single operation. This solution was reasonable because there was no reverse operation that created an entire syllable in a single operation, only one that parsed a syllable from existing segments. In this case, the most similar solution would be to prevent the unparsing of a syllable in a single step, but this solution would be blocking a reverse operation. The current model permits an operation V→ (v), so preventing the reverse (v)→V would be overly restrictive if other segments can be unparsed in a single step, as in the mapping (cvc)→(cv)C. If the unparsing operation exists in the model, it would be important for it to by symmetric with the parsing operation, without any ad hoc restrictions on its application. Currently, allowing the unparsing operation results in a very problematic inventory that violates implicational markedness. Without a more reasonable solution to this problematic inventory, adding unparsing to the set of operations would not be beneficial to the overall performance of the model.

3.4 Further considerations

In generating the main results discussed in section 3.2, some simplifying assumptions were made to make the simulations computationally tractable. In this section, I discuss some potential results when these features, particularly minor syllables and the Dep constraint, are included in the simulations. Each of these features introduces new challenges with problematic inventories. These problems will not be completely resolved here, but this will lay out the questions for future investigation.

3.4.1 Minor syllables

A potential structure that could be included in an analysis of syllable structure in an HS-based framework are minor syllables, as were a crucial component in Elfner's (2009) gradual syllabification, particularly as an intermediate step on the path to epenthesis. While I do not assume minor syllables as a possible structure in this typology, I will briefly discuss some of the possible implications of including minor syllables in an alternative typology. As it turns out, adding minor syllables to a syllable typology can cause a number of problematic interactions. In this section, I will describe one of these problematic interactions, which remains an area of study for future work. One interesting interaction occurs with minor syllables that potentially introduce a problematic inventory at a single level of the grammar. While this interaction has a potential solution with the addition of new operations, it provides a useful example of how additional structures, like minor syllables, introduce more complexity into the typology.

In this case, we consider the set of parsing operations previously discussed in the main analysis (parsing of a core syllable, projection of major syllable, adjunction into an onset or coda), plus one additional operation, parsing of a minor syllable. There is also one additional constraint, *Minor, that assigns a violation to each minor syllable. For this example, consider one possible ranking, ParseSeg>>Max>>Onset>>IdSyl>>*Minor>>NoCoda>>*Complex, which can result in the inventory {c, cv, ccv, cvc, ccvc, cvcc, ccvcc, v}. This inventory is unusual because it permits onsetless syllables and codas/complex clusters, but does not permit syllables with both of these features, namely (vc) and (vcc). As discussed in the previous section, we can permit this type of cumulative effect of markedness structures. What is odd here is that this is not the result of different levels with different rankings creating multiple types of marked structures. Instead, this is the result of an interesting interaction that can occur for minor syllables with this specific set of operations.

First, let us consider how (vc) syllables are excluded from the inventory under this ranking. From the /VC/ input, the first step of the derivation plays a critical role. From the ranking ParseSeg>>Onset>>Minor, the initial segment being parsed into a syllable is C rather than V.

(196) Step 1: parse minor syllable

/VC/	ParseSeg	Max	Onset	IdSyl	*Minor	NoCoda	*Complex
a. VC	W2				L		
b. (v)C	1		W1		L		
c. →V(c)	1				1		

The ranking of Onset>>*Minor compels the parsing of a minor syllable in the initial step because a violation of Onset is worse than a violation of *Minor. Even this initial step is novel with the introduction of a minor syllable related constraint and operation. Without them, the only parsing possibility in the initial step would be to parse the V into an onsetless syllable. This initial step will lead the derivation into a local minimum.

In the next step, the derivation continues by parsing the V into an onsetless syllable, which is permitted under the ranking of ParseSeg>>Onset.

(197) Step 2: Parse onsetless syllable

/V(c)/	ParseSeg	Max	Onset	IdSyl	*Minor	NoCoda	*Complex
a. V(c)	1				L		
b. →(v)(c)			W1		L		

While ParseSeg>>Onset permits the parsing of an onsetless syllable, there are no other harmonically improving options, so candidate (b) is selected as the winner. In the next step, the derivation converges on this form, as the ranking IdSyl>>Minor prevents resyllabification from (v)(c) to (vc).

(198) Converge

/(v)(c)/	ParseSeg	Max	Onset	IdSyl	*Minor	NoCoda	*Complex
a. →(v)(c)	1		1		1		
b. (vc)			1	W1	L	W1	

This end result is a little odd. Considering the ranking *Minor>>NoCoda, one might expect that the output [(vc)] would be preferred over [(v)(c)], so this interaction could be considered another example of a local minimum. The initial step of parsing a minor syllable prevents the standard path we would assuming for the parsing of a (vc) syllable under a different set of operations: VC → (v)C →(vc). Once the derivation has started on the path of parsing of minor syllable, it has no way of getting to the global maximum of (vc) under this ranking. One possibility might be to consider another operation that would permit a (vc) syllable after this initial step of parsing a minor syllable, such as V(c) →(VC). Such an operation had not been predefined in the parsing operations, so while adjoining a coda is permitted, adjoining a nucleus is not.

First let us consider how adding such an operation would add (vc) syllables into the inventory. Going back to step 2 in the previous derivation, once this operation is added, we have an additional candidate.

(199) Step 2: Adjoin nucleus

/V(c)/	ParseSeg	Max	Onset	IdSyl	*Minor	NoCoda	*Complex
a. V(c)	W1		L		W1	L	
b. (v)(c)			1		W1	L	
c. → (vc)			1			1	

The new candidate (c) permits parsing of the V into the minor syllable, or nucleus adjunction. This operation permits us to overcome the problem of the initial parsing of minor syllables into a VC syllable and the inability to resyllabify at a later stage.

Consider the related situation in which two minor syllables have been parsed in a similar manner, from an input of /VCC/. In this case, there is also a step of nucleus adjunction.

(200) Step 3: Nucleus adjunction

/V(c)(c)/	ParseSeg	Max	Onset	IdSyl	*Minor	NoCoda	*Complex
a. V(c)(c)	W1		L		W2	L	
b. (v)(c)(c)			1		W2	L	
c. → (vc)(c)			1		1	1	

The candidate with nucleus adjunction wins, as discussed in the previous derivation. However, in this case there is a remaining minor syllable that cannot be resyllabified, as before. Therefore, the derivation converges in the next step.

(201) Step 4: Converge

/(vc)(c)/	ParseSeg	Max	Onset	IdSyl	*Minor	NoCoda	*Complex
a. → (vc)(c)			1		1	1	L
b. (vcc)			1	W1	L	W2	W1

Under these assumptions, there is a remaining inventory of {c, cv, ccv, cvc, ccvc, cvcc, ccvcc, v, vc}, but not the desired inventory of {c, cv, ccv, cvc, ccvc, cvcc, ccvcc, v, vc, vcc}. To fix this problem, we must make one additional assumption, permitting minor syllables with more than one segment.

Under one such derivation, an additional operation of adjunction into a minor syllable would be permitted. In step 2, there is an additional candidate of V(cc), which is the winner.

(202) Step 2: Parse minor syllable

/V(c)C/	ParseSeg	Max	Onset	IdSyl	*Minor	NoCoda	*Complex
a. V(c)C	W2				1		
b. →V(cc)	1				1		
c. (vc)C	1		W1		L	W1	

In the following step, the adjunction of a nucleus into a complex minor syllable is permitted, thus bypassing the resyllabification problem.

(203) Step 3: Adjoin nucleus

/V(cc)/	ParseSeg	Max	Onset	IdSyl	*Minor	NoCoda	*Complex
a. V(cc)	W1		L		W1	L	
b. (v)(cc)			1		W1	L	
c. → (vcc)			1			1	

With the winner selected in this derivation, we round out the typology with a {c, cv, ccv, cvc, ccvc, cvcc, ccvcc, v, vc, vcc} inventory.

While adding (cc) syllables as a possible structure allows a complete typology of 16 desired inventories at the first level of the grammar, it introduces additional problems at the second level of the grammar. One such problematic inventory is {cc, cv}, which would be expected to contain a (c) syllable as well. One ranking that derives this output is ParseSeg>>NoCoda>>Onset>>*Complex>>IdSyl>>*Minor>>Max, which results in the mapping shown in (204).

(204) Problematic mapping

(cc) → (cc)

(c) → Ø

This mapping is similar to others discussed here because it involves a local minimum. The ranking of Minor>>Max at the second level permits deletion as a repair for minor syllables, as in the mapping (c) → Ø, but deletion for (cc) is not harmonically improving. The tableaux in (205) and (206) show how these mappings arise.

In the following tableau, the ranking of *Minor>>Max permits the deletion of the (c) input in a single step, resulting in the selection of the null output, shown in (205).

(205) Deletion harmonically improving

/(c)/	ParseSeg	NoCoda	Onset	*Complex	IdSyl	*Minor	Max
a. (c)						W1	L
b. → Ø							1

When the input is (cc), the same step of deletion is not harmonically improving. The violation profile of (cc) includes a single violation of *Minor and no additional markedness constraints. The deletion operation incurs a violation of Max, but does not remove any markedness violations, and thus is not harmonically improving, as shown in (206).

(206) Deletion not harmonically improving

/(cc)/	ParseSeg	NoCoda	Onset	*Complex	IdSyl	*Minor	Max
a. →(cc)						1	
b. (c)						1	W1

This interaction is similar to others involving problematic local minima. The globally optimal candidate is not available, so the derivation instead converges on a locally optimal candidate. In this case, as in others discussed in this chapter, this is not a desired result and here we would prefer to have the global optima as the output.

A solution for this case would likely involve a blocking of (c) deletion or permitting (cc) reduction or deletion to be harmonically improving. Unlike previous cases, there is not a straightforward solution to this problem. In a previous case, we prevented (v) from deleting in a single step. This was motivated by permitting deletion of a mora and segment in two separate operations, rather than a single one. Such a solution would be more difficult in this case because minor syllables by definition do not contain a mora, so there is no obvious way to split deletion into two separate operations. On the other hand, adding additional markedness violations to the (cc) syllable is more difficult than the case of complex codas. For example, adding two violations of *Minor would make deletion harmonically improving in this particular example, but would prevent (cc) syllables from being built in the first place because there would be no benefit over having two (c) syllables. Additionally, adding violations of a different markedness constraint would help for some rankings, but there would always be a ranking in the typology where that constraint was ranked below Max, thus not solving the problem completely.

This interaction caused by the introduction of minor syllables is rather interesting. It presents an additional case of an undesired local minimum predicted in Harmonic Serialism. There is the potential to work around this problem with the use of additional operations, so it does not necessarily exclude the possibility of using minor syllables in this analysis. However, the use of minor syllables does raise additional questions about the syllable structure and what types of operations are permitted, while the use of a computational typology brings up interesting examples that would otherwise be difficult to predict. While the minor syllables alone do not necessarily introduce any fatal flaws into the typology, as with other novel structures, they bring additional complexity that would need be dealt with. As will be discussed in the next section, precedents for epenthesis in HS-based frameworks rely on minor syllables, so either this complexity would need to be dealt with or an alternative theory of epenthesis in Stratal Harmonic Serialism would need to be proposed.

3.4.2 Dep

A major simplifying assumption made was the omission of the faithfulness constraint Dep from the simulations. There were two main reasons for this decision. First, there is an expectation that Max and Dep would generate different mappings, but lead to the same typology. Without additional grammars in the typology, having (n+1)! constraint rankings rather than n! would add significant additional computational complexity to the model. Second, the current theoretical assumption is that vowel epenthesis is dependent on minor syllables, with epenthetic vowels being added to existing minor syllables, following Elfner (2009). If minor syllables are not included in the model, adding Dep would require a new theoretical proposal for epenthesis, which there has been no motivations for up to this point.

To test some possible interactions with the Dep constraint, simulations were run with Dep and a set of epenthesis operations. The goal of these simulations was to gain some insight into the effects of Dep, rather than provide a complete typology with this constraint. For that reason, a limited set of epenthesis operations is used, including only those which are potentially harmonically improving: consonant epenthesis into an onset (v)→(cv), vowel epenthesis into a simple minor syllable (c)→(cv), and vowel epenthesis into a complex minor syllable (cc)→(cvc),(ccv). These operations are sufficient to investigate some of the complex interactions that are predicted.

Using these operations at a single level of the grammar, there were 18 resulting inventories in the typology, containing the 16 expected inventories, and two additional problematic inventories. One such problematic inventory was {cv, cc}, which was generated by the ranking ParseSeg>>NoCoda>>Onset>>*Complex>>Max>>IdSyl>>*Minor>>Dep. This inventory is problematic because it contains the syllable type (cc), but not (c). The surprising thing about this inventory is the ranking of *Minor>>Dep should permit minor syllables to be repaired by epenthesis, but the (cc) syllable is in the inventory.

First, let us consider why (c) is not in the inventory. In the first pass through the grammar, a minor syllable is parsed due to the ranking of ParseSeg>>*Minor.

(207) Step 1: Parse minor syllable

/C/	ParseSeg	NoCoda	Onset	*Complex	Max	IdSyl	*Minor	Dep
a. C	W1						L	
b. →(c)							1	

In the second pass through the grammar, there is a candidate generated by an epenthesis operation, candidate (b), that incurs a violation of Dep. Due to the ranking of *Minor>>Dep, the epenthesis candidate is the winner.

(208) Step 2: Epenthesis

/(c)/	ParseSeg	NoCoda	Onset	*Complex	Max	IdSyl	*Minor	Dep
a. (c)							W1	L
b. →(cv)								1

With an optimal (cv) syllable, the grammar converges on the third pass through the grammar.

(209) Step 3: Converge

/(cv)/	ParseSeg	NoCoda	Onset	*Complex	Max	IdSyl	*Minor	Dep
a. →(cv)								
b. (cvc)		W1						W1

With the use of Dep and epenthesis operations, the derivation of a (cv) syllable from a /C/ input is relatively straightforward. There are remaining (c) syllables in the inventory because under this ranking, epenthesis to a (cv) syllable will always be harmonically improving.

Now let us investigate why the (cc) syllable remains in the inventory, despite the *Minor>>Dep ranking. In the first step, a single minor syllable is parsed, resulting in an output of (c)C.

(210) Step 1: Parse minor syllable

/CC/	ParseSeg	NoCoda	Onset	*Complex	Max	IdSyl	*Minor	Dep
a. CC	W2						L	
b. →(c)C	1						1	
c. C	1				W1		L	

In the second step, the single minor syllable becomes a complex minor syllable.

(211) Step 2: Parse complex minor syllable

/(c)C/	ParseSeg	NoCoda	Onset	*Complex	Max	IdSyl	*Minor	Dep
a. (c)C	W1						1	
b. (c)(c)							2	
c. →(cc)							1	
d. (cv)C	W1							1

When selecting the candidate (cc), we crucially do not select the other candidates (c)(c) or (cv)C. Under the current constraint definitions, the syllable (cc) is always preferred over (c)(c) because it minimizes violations of *Minor and does not incur any additional markedness violations. The candidate (cv)C is not selected because the high ranking of ParseSeg compels parsing of all unparsed segments before other operations can apply.

In the next step, the derivation converges on (cc).

(212) Step 3: Converge

/(cc)/	ParseSeg	NoCoda	Onset	*Complex	Max	IdSyl	*Minor	Dep
a. →(cc)							1	
b. (cvc)		W1					L	W1
c. (ccv)				W1			L	W1
d. (vcc)			W1	W1			L	W1
e. (c)					W1		1	

The interesting aspect of this step is the failure of epenthesis to apply. While *Minor>>Dep, the markedness constraints NoCoda, *Complex, and Onset are all ranked above *Minor, and all possible epenthesis candidates would incur a fatal violation of one of these higher ranked markedness constraints. The globally optimal form, (cv)(cv), is not a possible candidate at any stage of the derivation, so instead the derivation converges on the locally optimal (cc).

This pattern of locally optimal outputs is one that has shown up in a number of different interactions, often between levels. In this case, the locally optimal output is selected at the first level of the grammar, so this result has implications for not just Stratal Harmonic Serialism, but for traditional Harmonic Serialism as well. The interaction of minor syllables and epenthesis in this theory shows some problematic results. There is still opportunity to redefine the set of operations to eliminate the remaining problematic inventories, so this is not fatal to the theory of gradual syllabification in Stratal Harmonic Serialism. The exact formulation of epenthesis operations and the status of minor syllables in the theory is a topic that warrants further study.

3.5 Discussion

In this section, I discuss some of the implications these results have for Stratal Harmonic Serialism, and Harmonic Serialism more generally.

3.5.1 Implications for Stratal Harmonic Serialism

From the results discussed in the previous sections, Stratal Harmonic Serialism does not predict grammars that are wildly overgenerative, but rather is restrictive in its predictions. Recall that none of the predicted inventories violate implicational markedness. With the assumption of Gradual Syllabification and the constraints and operations presented here, the theory does predict cumulativity effects, or additive markedness, in a subset of the grammars in the typology. These effects occurred in syllable inventories like {cv, cvc, v}, where the marked structures violating NoCoda and *Onset were permitted, but the syllable (vc) violating both of these constraints in not present. While this does not require cumulativity to be present in all languages, there must be evidence of cumulativity for this prediction to be empirically accurate.

With regards to syllable structure, these types of cumulative effects are found in acquisition data. One study (Levelt, Schiller, and Levelt 2000) found that the acquisition of syllable types in Dutch children followed the paths shown in Figure 5.

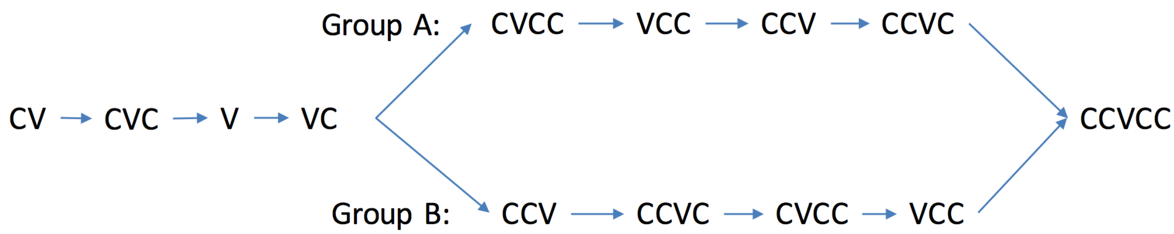

Figure 5: Acquisition data (Levelt et al. 2000)

These results show that more complex syllable types, like (ccvcc), are acquired after children have already acquired the marked features that should permit the cumulative effect of these features. For example, there is a stage during which children have acquired codas and onsetless syllables, but not yet acquired the (vc) syllable. At this stage, their syllable inventory is {cv, cvc, v}, which is the same as a syllable inventory predicted by Stratal Harmonic Serialism. Child language data may not be the best evidence for empirical claims about adult grammars, with this pattern accounted for under different models (Albright, Magri, and Michaels 2008).

Outside the domain of syllable structure, cumulativity effects are discussed in the literature on Constraint Conjunction (Ito and Mester 1996; Lubowicz 2002; Padgett 2002; Smolensky 1995, 2006) and Harmonic Grammar (Pater 2009). There are many documented interactions where two marked structures are permitted, but the violation of both is not. In the case of Constraint Conjunction, this is explained through a highly ranked constraint that incurs a violation when two marked structures occur simultaneously. In Harmonic Grammar, this is modelled by Gang-up Effects, where the sum of two weighted constraints violations disqualifies the cumulative candidate. In the case of Stratal Harmonic Serialism, cumulativity is the result of two levels with different constraint rankings. Not all such cases of cumulativity can be covered by the current theory due to the leveling of the first level results by the second level. The presence of cumulativity effects in the resulting typology may not problematic for the theory because they are a common phenomenon, but more work is needed to determine the overlap between cumulativity effects predicted by Stratal Harmonic Serialism and those found in natural language.

3.5.2 Implications for Harmonic Serialism

In addition to the implications for Stratal Harmonic Serialism, we can also extend some of these results to apply to Harmonic Serialism more generally. A major point here has been that some types of locally-optimal interactions are not desired. While the gradual property of Harmonic Serialism that permits locally-optimal over globally-optimal ones is beneficial in many cases (Elfner 2009; McCarthy 2008, 2009), these results demonstrate the potential for undesirable local mappings.

Many of the local mappings discussed in this chapter follow a similar pattern with regard to constraint violation schemas. Let us re-examine two such cases, the formulation of NoCoda discussed in section 3.3.1 and the deletion of onsetless syllables discussed in section 3.3.2. In particular, we are focusing on the two relevant inputs, and the two relevant candidates for each.

First, recall the problematic locally-optimal set of mappings that led to the new formulation of NoCoda, shown in (213).

(213) Formulation of NoCoda: Mappings leading to unattested inventory

Input 1: /(cvc)/ → (cv)

Input 2: /(cvcc)/ → (cvcc)

One ranking used to generate these mappings was NoCoda(old)>>Max>>*Complex>> ParseSeg>>Onset>>IdSyl. For now, I will only focus on the important component of this ranking, NoCoda(old)>>Max>>*Complex. The tableaux in (214)-(215) show the violation profile of the two inputs and the relevant candidates, faithful and unfaithful.

(214) No Simple Codas with NoCoda(old)

Input 1	/(cvc)/	NoCoda(old)	Max	*Complex
Faith	a. (cvc)	W1	L	
Unfaith	b. →(cv)		1	

(215) Complex codas with NoCoda(old)

Input 2	/(cvcc)/	NoCoda(old)	Max	*Complex
Faith	a. →(cvcc)	1		1
Unfaith	b. (cvc)	1	W1	L

Now recall the mappings that motivated the prohibition of deletion of onsetless syllables, shown in (216).

(216) Deletion of onsetless syllables: Mappings leading to unattested inventory

Input 1: /(v)/ → Ø

Input 2: /(vc)/→ (vc)

A ranking that generates this mapping is

Onset>>Max>>NoCoda>>*Complex>>ParseSeg >>IdSyl. For now, we will just focus on the ranking of the relevant constraints, Onset>>Max>>NoCoda. The tableaux for the two inputs and their relevant candidates are shown in (217)-(218).

(217) No onsetless syllables

Input 1	/(v)/	Onset	Max	NoCoda
Faith	a. (v)	W1	L	
Unfaith	b. →Ø		1	

(218) Onsetless syllables with codas

Input 2	/(vc)/	Onset	Max	NoCoda
Faith	a. →(vc)	1		1
Unfaith	b. (v)	1	W1	L

From these two different cases of locally-optimal mappings, we can abstract the constraints to their markedness and faithfulness categories, leading to a shared M1>>F>>M2 rankings. The constraint violation profiles for the relevant inputs and candidates of these two cases are identical, as shown in (219).

(219) Schema for potentially problematic local mappings

Input 1	M1	F	M2
Faith	W1	L	
→ Unfaith		1	
Input 2	M1	F	M2
→ Faith	1		1
Unfaith	1	W1	L

In this schema, there are three constraints in the ranking M1>>F>>M2. The Faith candidate of Input1 and Input 2 violate M1, and the Faith candidate of Input 2 also has a violation of M2. Input1's Unfaith candidate is able to satisfy M1 with a violation of F, but Input 2 has no such candidate and is blocked from doing this due to some property of M2. Because Input 2 does not have any harmonically improving unfaithful candidates, the Faith candidate is selected.

We have seen such a case occur with between levels in Stratal Harmonic Serialism, with structures like (cvcc) and (vc) that can be built under one ranking, but are difficult to deconstruct under another, resulting in problematic inventories containing {cvcc, *cvc} and {vc, *v}.

In Harmonic Serialism, there is the potential for such interactions to occur as well. These cases are more difficult to construct because they cannot be due to structure, which must be built, but rather, would need to occur underlyingly through Richness of the Base. There may well be cases that fall under this general schema to produce problematic locally-optimal mappings in Harmonic Serialism.

3.6 Conclusion

The simulations presented here demonstrate that Stratal Harmonic Serialism with Gradual Syllabification has the potential to predict a syllable structure typology that is neither overgenerative nor undergenerative. With a specific set of constraints and operations, the framework can produce a set of inventories that do not violate implicational markedness, but still include the expected set of inventories. The alternative simulations presented here show the reasons for some crucial decisions, and also demonstrate the potential for problematic locally-optimal interactions. In chapter 4, I will discuss how these results relate to phonology-morphology interactions in the model and in natural language.

Chapter 4 Phonology-Morphology interactions

The interaction between phonology and morphology plays a key role in Stratal Harmonic Serialism, not only through a key element of the framework, morphological levels, but in the effects that the introduction of morphological material has on the building of prosodic structure and surface forms. In this chapter, I discuss three areas in which phonology-morphology interactions surface in the framework: cumulativity, asymmetry, and opacity.

4.1 Cumulativity

Cumulativity here refers to the effect of syllable type inventories that are created by adding additional syllables to one of the “standard” syllable inventories (these 8 inventory types are presented in Chapter 3). These are inventories in which some syllable types appear to be missing, but do not violate implicational markedness. In Stratal Harmonic Serialism, cumulativity can be an effect of levels or an effect of affixes, both of which are cases of a morphology-phonology interaction.

4.1.1 Cumulativity as an effect of levels

Cumulativity as an effect of levels is the result of constraint reranking at a later level. In the syllable inventory typology, two additional inventories of this type are generated by the model, [cv, cvc, ccv, ccvc] and [cv, cvc, ccv, ccvc, v, vc]. Each of these inventories is missing some syllable types preventing them from being a complete syllable inventory as seen generated by the first level of the grammar. The inventory [cv, cvc, ccv, ccvc] contains the marked structures of codas and complex clusters, but is lacking the syllables cvcc and ccvcc. Similarly, [cv, cvc, ccv, ccvc, v, vc] has codas, complex clusters, and onsetless syllables, but is lacking cvcc, ccvcc, and vcc.

When cumulativity occurs as an effect of levels, there is a new constraint ranking, but no new affixal material, so changes apply to the existing structures, which in the domain of syllabification presents as resyllabification.

*4.1.1.1 ParseSeg>>Onset>>Max>>*Complex>>NoCoda>>IdSyl*

4.1.1.1.1 [cv, ccv, cvc, ccvc]

Under the ranking of ParseSeg>>Onset>>Max>>*Complex>>NoCoda>>IdSyl, the cumulative inventory [cv, ccv, cvc, ccvc] is created from the input inventory [cv, ccv]. At the boundaries of these syllables the syllable types cvc and ccvc are created through resyllabification.

In tableaux (220)-(221), a cvc syllable is created from the input /(cv)(ccv)/. Resyllabification occurs in the first step at the second level of the grammar.

(220) Step 1: Resyllabification to (cvc)

/(cv)(ccv)/	ParseSeg	Onset	Max	*Complex	NoCoda	IdSyl
a. (cv)(ccv)				W1	L	L
b. → (cvc)(cv)					1	1
c. (cv)(cv)			W1		L	L

The ranking of Complex>>NoCoda prefers codas over complex clusters. In the winning candidate (b), the existing complex onset is repaired via resyllabification to a coda and simple onset, incurring a violation of IdSyl. Deletion of the complex cluster in candidate (c) is not harmonically improving due to Max>>Complex.

(221) Step 2: Convergence

/(cvc)(cv)/	ParseSeg	Onset	Max	*Complex	NoCoda	IdSyl
a. →(cvc)(cv)					1	
b. (cv)(cv)			W1		L	

There are no remaining harmonically improving operations in the second step, so the derivation converges, adding a new cvc syllable to the inventory.

In tableaux (222)-(223), a ccvc syllable is created through resyllabification from the input /(ccv)(ccv)/.

(222) Step 1: Resyllabification to (ccvc)

/(ccv)(ccv)/	ParseSeg	Onset	Max	*Complex	NoCoda	IdSyl
a. (ccv)(ccv)				W2	L	L
b. → (ccvc)(cv)				1	1	1
c. (cv)(ccv)			W1	1	L	L
d. (ccv)(cv)			W1	1	L	L

As in the previous derivation, resyllabification in the winning candidate (b) occurs due to the preference for codas over complex clusters and the low ranking of IdSyl. In this case, the resyllabified consonant moves into a ccv syllable, so a ccvc syllable is created.

(223) Step 2: Convergence

/(ccvc)(cv)/	ParseSeg	Onset	Max	*Complex	NoCoda	IdSyl
a. → (ccvc)(cv)				1	1	
b. (cvc)(cv)			W1	L	1	

In the second step, the derivation converges, with the new ccvc syllable in the final output.

The set of inputs and constraint ranking in this grammar results in a cumulative inventory [cv, cvc, ccv, ccvc]. While complex clusters are not preferred in this ranking, they cannot be deleted due to Max>>*Complex. Complex onsets are resyllabified into codas under the right conditions, namely when a syllable with a complex onset immediately follows a syllable without a coda. Resyllabification results in new cvc and ccvc syllables, but new complex codas are not created, so the inventory is lacking the otherwise expected cvcc and ccvcc, resulting in a cumulative inventory.

4.1.1.1.2 [cv, ccv, cvc, ccvc, v, vc]

The second cumulative inventory that shows an effect of levels can be generated with the same constraint ranking and a different set of inputs, [cv, ccv, v]. As in the previous case, the syllable types cvc and ccvc are created through resyllabification, but in this case there is an additional syllable type, vc. The derivation of the vc syllable is shown in tableaux (224)-(225).

(224) Step 1: Resyllabification to (vc)

/(v)(ccv)/	ParseSeg	Onset	Max	*Complex	NoCoda	IdSyl
a. (v)(ccv)		1		W1	L	L
b. → (vc)(cv)		1			1	1
c. (v)(cv)		1	W1		L	L

In the first step, a consonant from the complex onset resyllabifies into the coda of the preceding onsetless syllable in the winning candidate (b). Reduction of the complex cluster through deletion in candidate (c) is not harmonically improving due to Max>>*Complex.

(225) Step 2: Convergence

/(vc)(cv)/	ParseSeg	Onset	Max	*Complex	NoCoda	IdSyl
a. → (vc)(cv)		1			1	
b. (v)(cv)		1	W1		L	

With no remaining harmonically improving operations, the derivation converges in the next step, with a new vc syllable.

As in the related cumulative inventory, the additional syllable types cvc, ccvc, and vc were created via resyllabification. The new ranking at the second level permits the creation of new codas, but not the deletion of existing complex clusters. This inventory differs from the previous in only the addition of a v syllable, from the input, and a vc syllable, created through resyllabification from the v input. Despite the presence of both codas and complex clusters, this inventory [cv, ccv, cvc, ccvc, v, vc] lacks all syllable types with complex codas, making it cumulative.

4.1.2 Cumulativity as an effect of affixes

Cumulativity as an effect of affixes occurs due to the parsing of new affixal material at the second level of the grammar. With a new constraint ranking at the second level, new syllables that may not have been possible at the previous level can now be parsed. The addition of new syllable types to old inventories can result in cumulative inventories.

4.1.2.1 [cv, cvc, v]

The inventory [cv, cvc, v] exhibits cumulativity as an effect of affixes. It is generated from the input inventory [cv, cvc] under the ranking *Complex>>Max>>NoCoda>>ParseSeg>>Onset>>IdSyl. In this case, the existing codas cannot be deleted due to Max>>NoCoda, but new codas cannot be parsed, due to NoCoda>>ParseSeg. Additionally, new (v) syllables are parsed from affixes under ParseSeg>>Onset, but (vc) syllables cannot be created because of the constraints against new codas. This results in a cumulative inventory that contains the marked structures of codas and onsetlessness, but not together in the same syllable.

For the derivation of a new onsetless syllable, the input begins with a V affix undergoing syllable projection:

(226) Step 1: Project syllable

/V(cvc)/	*Complex	Max	NoCoda	ParseSeg	Onset	IdSyl
a. V(cvc)			1	W1	L	
b. -> (v)(cvc)			1		1	
c. (cvc)		W1	1		L	
d. V(cv)		W1	L	W1	L	

The winning candidate (b) parses a new onsetless syllable under ParseSeg>>Onset. Note that the deletion candidates (c) and (d) are not harmonically improving.

In the next step, the derivation converges:

(227) Step 2: Converge

/V(cvc)/	*Complex	Max	NoCoda	ParseSeg	Onset	IdSyl
a. ->(v)(cvc)			1		1	
b. (vc)(vc)			W2		W2	W1
c. (v)(cv)		W1	L		1	

Losing candidates exhibiting resyllabification and deletion are not harmonically improving, so the faithful candidate wins.

This ranking adds new v syllables, resulting in the [cv, cvc, v] inventory, which is cumulative because it lacks syllables that combine the marked structures of codas and onsetlessness, namely vc.

4.1.2.2 [cv, cvc, ccvc, ccv, v]

This inventory begins with the set of inputs [cv, cvc] and creates three additional new syllable types through parsing, under the ranking Max>>NoCoda>>ParseSeg>>Onset>>*Complex>>IdSyl. This ranking permits the creation of new onsetless syllables and new complex onsets, not no new codas, simple or complex.

The tableaux in (228)-(229) show the derivation of the new ccvc syllable type, from the input /C(cvc)C/.

(228) Step 1: Parse complex onset

/C(cvc)C/	Max	NoCoda	ParseSeg	Onset	*Complex	IdSyl
a. C(cvc)C		1	W2		L	
b. →(ccvc)C		1	1		L	
c. C(cvcc)		W2	1		1	

In the first step, an unparsed C is parsed into a complex onset of the existing cvc syllable in the winning candidate (b), due to the ranking ParseSeg>>*Complex. Parsing the other unparsed C into a complex coda, as in candidate (c), is not harmonically improving due to NoCoda>>ParseSeg.

(229) Step 2: Converge

/(ccvc)C/	Max	NoCoda	ParseSeg	Onset	*Complex	IdSyl
a. → (ccvc)C		1	1		1	
b. (ccvcc)		W2	L		W2	
c. (ccvc)	W1	1	L		1	

With no remaining harmonically improving operations, the derivation converges in the next step, adding a new ccvc syllable to the inventory.

Tableaux (230)-(232) show the derivation of the new v and ccv syllables from the input /VC(cv)/.

(230) Step 1: Parse complex onset

/VC(cv)/	Max	NoCoda	ParseSeg	Onset	*Complex	IdSyl
a. VC(cv)			W2		L	
b. →V(ccv)			1		1	
c. (v)C(cv)			1	W1	L	

In the first step, the unparsed C is parsed into the existing cv syllable in candidate (b) due to the ranking ParseSeg>>*Complex. At this step, candidate (c), in which an onsetless syllable is parsed due to ParseSeg>>Onset, is harmonically improving, but not the optimal candidate, due to Onset>>*Complex.

(231) Step 2: Parse onsetless syllable

/V(ccv)/	Max	NoCoda	ParseSeg	Onset	*Complex	IdSyl
a. V(ccv)			W1	L	1	
b. →(v)(ccv)				1	1	

The remaining unparsed V is parsed into an onsetless syllable in the second step. In the next step the derivation converges.

(232) Step 3: Converge

/(v)(ccv)/	Max	NoCoda	ParseSeg	Onset	*Complex	IdSyl
a. → (v)(ccv)				1	1	
b. (v)(cv)	W1			1	L	
c. (vc)(cv)		W1		1	L	W1

With no remaining harmonically improving steps, the derivation converges on (v)(ccv), which contains the remaining two additional syllable types in the cumulative inventory, v and ccv.

4.1.2.3 [cv, cvc, ccv, cvcc, ccvc, ccvcc, v]

This cumulative inventory is created when the inventory of inputs is [cv, cvc, ccv, cvcc, ccvc, ccvcc] and the ranking can parse new onsetless syllables, but not new codas. One such ranking is Max>>NoCoda>>ParseSeg>>Onset>>*Complex>>IdSyl. The tableaux in (233)-(235) show how a onsetless syllable is parsed, and the failure to parse a new coda.

(233) Step 1: Parse complex onset

/VC(cvc)/	Max	NoCoda	ParseSeg	Onset	*Complex	IdSyl
a. VC(cvc)		1	W2		L	
b. ->V(ccvc)		1	1		1	
c. (v)C(cvc)		1	1	W1	L	

In the first step, the unparsed C is parsed into a complex onset, due to ParseSeg>>*Complex in candidate (b). Candidate (c), which parses V into an onsetless syllable, is harmonically improving but not the optimal candidate at this step.

(234) Step 2: Parse onsetless syllable

/V(ccvc)/	Max	NoCoda	ParseSeg	Onset	*Complex	IdSyl
a. V(ccvc)		1	W1	L	1	
b. ->(v)(ccvc)		1		1	1	

In the second step, the remaining V is parsed into a new syllable, due to ParseSeg>>Onset. In the next step, the derivation converges.

(235) Step 3: Converge

/(v)(ccvc)/	Max	NoCoda	ParseSeg	Onset	*Complex	IdSyl
a. ->(v)(ccvc)		1		1	1	
b. (v)(ccv)	W1	L		1	1	
c. (vc)(cvc)		W2		1	L	W1

With no remaining harmonically improving steps, the derivation converges, with the new v syllable added to the cumulative inventory. Resyllabification of the complex onset into a coda, as seen in candidate (c), is not harmonically improving, due to NoCoda>>*Complex.

4.1.2.4 [cv, cvc, ccv, cvcc, ccvc, ccvcc, v, vc]

As in the previous case, this cumulative inventory is created when the inventory of inputs is [cv, cvc, ccv, cvcc, ccvc, ccvcc], but here the ranking permits the parsing of new codas in addition to onsetless syllables, resulting in the additional syllable type vc. Tableaux (236)-(238) show the derivation of a vc syllable from the input /VC(cvc)/ under the ranking Max>>*Complex>>ParseSeg>>NoCoda>>Onset>>IdSyl.

(236) Step 1: Project syllable

/VC(cvc)/	Max	*Complex	ParseSeg	NoCoda	Onset	IdSyl
a. VC(cvc)			W2	1	L	
b. ->(v)C(cvc)			1	1	1	
c. V(ccvc)		W1	1	1		

In the first step, the winning candidate (b) parsed the onsetless V into a new syllable due to PaseSeg>>Onset. Parsing of the unparsed C into a complex onset of the following syllable in candidate (c) is not harmonically improving due to *Complex>>ParseSeg.

(237) Step 2: Adjoin coda

/(v)C(cvc)/	Max	*Complex	ParseSeg	NoCoda	Onset	IdSyl
a. (v)C(cvc)			W1	L1	L	
b. ->(vc)(cvc)				2	1	
c. (v)(ccvc)		W1		L1	1	

In the second step, the remaining unparsed C can be parsed into the coda of the newly created v syllable due to ParseSeg>>NoCoda, as seen in the winning candidate (b). Again, parsing that C into a complex onset, as in candidate (c) in not harmonically improving.

(238) Step 3: Converge

/(vc)(cvc)/	Max	*Complex	ParseSeg	NoCoda	Onset	IdSyl
a. ->(vc)(cvc)				2	1	
b. (v)(cvc)	W1			L1	1	

The derivation converges in the next step, with a new vc syllable added to the cumulative inventory.

4.1.3 Discussion

The results in this section show the different ways that cumulativity is derived in syllable structure inventories, as an effect of levels and affixes. The novel components of Stratal Harmonic Serialism predict additional syllable inventories beyond those predicted by parallel OT and strict Harmonic Serialism.

The morphological effects of levels are seen most clearly in cumulativity, due to the creation of new syllable inventories at the second level. However, non-cumulative syllable inventories also show evidence of morphological effects. For example, the non-cumulative inventory {cv, cvc} can be derived in multiple ways. In a minimally changing case, the output inventories from both the first and second levels are {cv, cvc}, with either identical or not significantly different constraint rankings between levels. In another case, the output inventories are different, for example {cv} at the first level and {cv, cvc} at the second level. While surface outputs in the syllable inventory are the same, these two derivations represent different morphological structures. In the constant case, stems can have codas, while in the changing case, stems do not have codas, only affixes do.

The syllable structure cumulativity predicted by Stratal Harmonic Serialism is always an effect of morphology, through the reranking of constraints between levels, the introduction of new material, or some combination of the two, mechanisms not present in strict Harmonic Serialism or parallel OT. The generation of additional syllable inventories, as well the differences in stems and affixes between levels, is a result of some key differences with what occurs at the second level compared to the first. One main difference is Richness of the Base. While the model assumes a rich set of inputs at the first level and affixes at the second level, stems at the second level are much sparser, as they have been filtered through the first level. A second key difference is structure. The inputs to the second level have existing syllable structure.

4.2 Asymmetry

Cumulativity effects are a key prediction of the Stratal Harmonic Serialism model, as discussed in the previous section. In this section, I discuss syllable asymmetries, or cases where syllabification patterns differ due to morphology-phonology interactions, and their relationship to cumulativity effects. Here I present some cases of syllable asymmetries in natural language and discuss their interpretation in Stratal Harmonic Serialism.

4.2.1 Spanish

Spanish resyllabification occurs across prefix and word boundaries to produce a syllabification pattern that is dispreferred within words. Within monomorphemic words, complex onsets are preferred over coda-onset sequences, whereas coda-onset sequences are preferred across morpheme and word boundaries (Face 2002). Within words, Spanish has a preference for complex onsets over codas, as seen by the syllabification of [a.βlaɾ] in (239).

(239) Complex onsets in Spanish (Face 2002)

a.βlaɾ 'to speak'

pe.re.ɣɾi.no 'pilgrim'

However, across morpheme boundaries there is a different pattern. While resyllabification of a coda consonant into a simple onset is possible across the prefix boundary in (240), the coda does not resyllabify across a prefix boundary to form a complex onset in (240).

(240) Resyllabification across prefix boundaries (Face 2002)

a. i.n | es.pe.ɾa.ðo 'unexpected'

su.β | a.kwa.ti.ko 'submarine (adj.)'

b. að | .ri.sar 'to right (nautical)'

suβ | .ra.jar 'underline'

The same pattern is found at word boundaries. A coda resyllabifies as a singleton onset in (241), but not in (241).

(241) Resyllabification across word boundaries (Face 2002)

a. klu.β | e.le.ɣan.te 'elegant club'

u.n | e.le.fan.te 'an elephant'

b. kluβ | .lin.do 'beautiful club'

be.nið | .ra.pi.ðo 'come pl. imper. quickly'

The syllable asymmetry shown in this case is a preference for complex onsets over codas at the word level, but a preference for codas over derived complex onsets at the phrase level. In Face's (2000) account, the difference in syllabification patterns is accounted for with alignment constraints.

In Stratal Harmonic Serialism, the difference in syllabification patterns is accomplished through ranking of constraints between the word and phrase levels.

At the word level, the grammar parses segments into syllables. The syllabification of [a.βlaɾ] shows the general word-level syllable types: onsets are not required, complex clusters and codas are permitted, and with intermediate consonant clusters, complex onsets are preferred to codas. Syllables are parsed gradually beginning with core syllabification, followed by syllabification of the remaining unparsed segments. The tableau in (242) shows harmonic improvement for each step of syllabification at the word level under the proposed ranking[16]:

(242) Harmonic improvement for syllabification at word level

/ aβlaɾ/	ParseSeg	NoCoda	*Complex	Onset	IdSyl
aβ(la)ɾ	3				
(a)β(la)ɾ	2			1	
(a)(βla)ɾ	1		1	1	
(a)(βlaɾ)		1	1	1	

With the final output of the word level, this constraint ranking shows a preference for syllables with complex onsets over codas. The pattern seen at the phrase level shows the reverse preference.

At the phrase level, resyllabification occurs to repair onset violations, but does not resyllabify codas into complex onsets. With a reranking of a faithfulness constraint governing syllable identity, resyllabification is permitted in (243), but not in (244):

[16] No deletion candidates are considered in this example, so the ranking of Max is not included here.

(243) Resyllabification across word boundaries

/ (kluβ) | (e)(le)(γan)(te) /	ParseSeg	Onset	IdSyl	NoCoda	*Complex
a. (kluβ) | (e)(le)(γan)(te)		W1	L	2	1
b. → (klu) | (βe)(le)(γan)(te)			1	1	1

(244) No resyllabification across word boundaries

/ (kluβ) | (lin)(do) /	ParseSeg	Onset	IdSyl	NoCoda	*Complex
a. →(kluβ) | (lin)(do)				2	1
b. (klu) | (βlin)(do)			W1	L1	L

In (243), the ranking Onset>>IdSyl permits resyllabification, while in (244), IdSyl>>NoCoda prevents resyllabification. The ability of this framework to account for these facts is due to the distinct levels of the grammar. At the phrase level, the grammar does not resyllabify existing complex onsets, seen in (245):

(245) No resyllabification within words

/ (a)(βlaɾ) /	ParseSeg	Onset	IdSyl	NoCoda	*Complex
a. →(a)(βlaɾ)		1		1	1
b. (aβ)(laɾ)		1	W1	W2	L

The case of syllable asymmetries in Spanish resyllabification offers evidence for the framework of Stratal Harmonic Serialism. Though not a case of cumulativity effects, it shows different patterns marked by constraint reranking between morphological levels. In strictly parallel OT, additional mechanisms, such as alignment constraints, are needed to account for this data.

4.2.2 Donceto Italian

Subject clitics in Donceto Italian exhibit a different syllabification pattern from prosodic words. While complex clusters are permitted in both onsets and codas, subject clitics undergo epenthesis where they would otherwise create derived complex clusters (Cardinaletti and Repetti 2009). The surface forms of prosodic words like [tri] ‘three’ and [ust] ‘August’ contain complex clusters, as shown in (246), while those same sequences are not permitted when the mono-consonantal clitic /t/ would create the same sequence, as shown in (246).

(246) Syllable asymmetry in Donceto Italian (Cardinaletti and Repetti 2009)

a.	tri	‘three’	ust	‘August’
b.	ət-rõːf	‘you:sg snore’	pas-ət	‘do you:sg pass?’
	*t-rõːf		*pas-t	

This clitic is underlying /t/ rather than /ət/ due to forms like the one in (247).

(247) Mono-consonantal clitic

t-ɛ buˈviːd	*ət-ɛ buˈviːd	‘you:sg have drunk’
ɛ-t buˈviːd	*ɛ-ət buˈviːd	‘have you:sg drunk?’

The syllable asymmetry shown here is a acceptance of complex onsets at the word level, but a dispreference for derived complex onsets at the phrase level. Cardinaletti and Repetti (2009) account for this data using alignment constraints referring to the different prosodic boundaries.

In Stratal Harmonic Serialism, the analysis would follow a similar pattern as the Spanish resyllabification in the previous example. There are different constraint rankings between the word and phrase level, in order to permit the different syllabification patterns found in monomorphemic and cliticized words. This example involves vowel epenthesis, an operation not included in the simulations in Chapter 3, so I do not include a complete analysis of this case in Stratal Harmonic Serialism, though the existence of this pattern in natural language is important to note.

4.2.3 Tzutujil

The syllable structure of clitics in Tzutujil are less restricted than other forms. Glottal stop insertion in word-initial onsetless syllables is a common process in Mayan languages (Bennett 2016). While a glottal stop is usually inserted in onsetless syllables, this does not apply to vowel initial clitics (Dayley 1985). In monosyllabic words, a glottal stop is always inserted in onsetless syllables, as shown in (248), though this process is optional in polysyllabic words, as shown in (248).

(248) No onsetless syllables (Dayley 1985)

a.	ak’	[ʔak’]	‘chicken’
	ooj	[ʔoːx]	‘avocado’
b.	iteel	[(ʔ)ɪt͡seːl̥]	‘bad, ugly’
	elaq’	[(ʔ)ɛlaq’]	‘robbery’

However, the vowel-initial absolutive proclitic and directional enclitic do not undergo glottal stop insertion, as shown in (249).

(249) Onsetless clitics (Dayley 1985)

in winaq [ɪn βɪnaqʰ] 'I am a person'

xel eel [ʃɛl̥ eːl̥] 'he went out'

The syllable asymmetry shown here is the dispreference for onsetless syllables at the word level, but an acceptance of them in derived forms at the phrase level.

In Stratal Harmonic Serialism, the constraints governing onsetless syllables and repair via glottal stop insertion would differ between the word and phrase levels. While I do not cover epenthesis and Dep in the chapter 3 typology, the epenthesis of an onset consonant is less contentious than a syllable nucleus, so I will provide a sketch of the SHS analysis here. At the word level, the mapping /ak'/ → [ʔak'] will pass through an intermediate stage of an onsetless syllable, first projecting a syllable in (250):

(250) Step 1: Syllable projection at the word level

/ak'/	ParseSeg	Onset	Dep	NoCoda
a. ak'	W2	L		
b. → (a)k'	1	1		

In this step the ranking ParseSeg>>Onset allows parsing of an onsetless syllable, though this form does not surface at the word level. In the next step, the coda consonant is adjoined, to satisfy the remaining violation of ParseSeg, shown in (251):

(251) Step 2: Coda adjunction at the word level

/(a)k'/	ParseSeg	Onset	Dep	NoCoda
a. (a)k'	W1	1		L
b. → (ak')		1		1

In the final step before converging, a glottal stop in inserted due to the ranking Onset>>Dep, shown in (252):

(252) Step 3: Glottal stop insertion at the word level

/(ak')/	ParseSeg	Onset	Dep	NoCoda
a. (ak')		W1	L	1
b. → (ʔak')			1	1

The ranking changes at the phrase level, where we need to account for the mapping (with intermediate form input) /ɪn-(βɪ)(naqʰ)/ → [ɪn βɪnaqʰ]. As this mapping no longer requires glottal stop insertion to repair onsetless syllables, the ranking changes from Onset>> Dep to Dep>> Onset. The parsing of the clitic occur as in the previous level, shown in (253) and (254):

(253) Step 1: Syllable projection at the phrase level

/ɪn-(βɪ)(naqʰ)/	ParseSeg	Dep	Onset	NoCoda
a. ɪn(βɪ)(naqʰ)	W2		L	1
b. → (ɪ)n(βɪ)(naqʰ)	1		1	1

(254) Step 2: Coda adjunction at the phrase level

/(ɪ)n-(βɪ)(naqʰ)/	ParseSeg	Dep	Onset	NoCoda
a. (ɪ)n(βɪ)(naqʰ)	W1		1	L1
b. → (ɪn)(βɪ)(naqʰ)			1	2

After the clitic has been parsed, the derivation converges, as the step of glottal stop insertion is not harmonically improving, shown in (255):

(255) Step 3: No glottal stop insertion at the phrase level

/(ɪn)-(βɪ)(naqʰ)/	ParseSeg	Dep	Onset	NoCoda
a. → (ɪn)(βɪ)(naqʰ)			1	L1
b. (ʔɪn)(βɪ)(naqʰ)		W1	L	2

In the case of Tzutujil, we have another example of syllable structure asymmetry, where the syllable types created at the word level are different from those at the phrase level. In this case, the restriction on onsetless syllables at the word level is not enforced at the phrase level, but those glottal stops inserted at the word level still surface in forms after the phrase level. This shift from a less permissive to more permissive level leaves different structures present in surface forms, based on morphological composition.

4.2.4 Yine (Piro)

A syllable type asymmetry is found in Yine, where some derived clusters are permitted is the creation of initial CCC clusters, which are otherwise not found in the language. An active process of vowel deletion is blocked when it would otherwise create CCC clusters, which are permitted word-initially in derived environments (Zimmerman 2013). There is a process of vowel deletion, which occurs at morpheme boundaries of some affixes, as shown in (256)a. However, when this vowel deletion would create a medial CCC cluster, vowel deletion is blocked, as shown in (256)b.

(256) Vowel deletion (Matteson 1965; Zimmermann 2013)

a.	/neta-lu/	netlu	'I see him'
	/walapu-ni/	walapni	'the past summer'
b.	/terka-lu/	terkalu	'she washes it'
	/kpitxnu-tsi/	kpitxnutsi	'thick, please'

The restriction against CCC clusters does not apply everywhere is the language. CCC clusters are created by monoconsonantal prefixes, as seen in (257).

(257) Initial CCC cluster (Lin 1997; Zimmermann 2013)

a. /n-knojate/	nknojate	'my turtle'
b. /p-knojate/	pknojate	'your turtle'

The syllable asymmetry here is a dispreference for medial derived CCC clusters, but acceptance of edge CCC clusters that contain monoconsonantal prefixes; edge clusters do not appear in monomorphemic words.

The account of this data in Stratal Harmonic Serialism would involve locating the different restrictions on syllabification patterns at different levels of the grammar. While I do not specifically account for triconsonantal clusters in the typology, one could extend the current constraint set to permit them.

4.2.5 English

English syllable structure is systematically different at the edges. Medial consonant clusters are more restrictive than those that appear word-finally beyond what can be explained by appendices, which is argued to be a more restrictive rime at level 1 compared to level 2 within a lexical phonology framework (Borowsky 1989). Borowsky argues that because vowel shortening applies at level 1 in derived words, the relative absence of medial tautosyllabic VVC sequences is due to a VX rime at level 1 versus VXC at level 2 (e.g. the presence of words like *apron* but not **anpron*). Medial VVC rimes that are followed by a homorganic consonant (e.g. *chamber*) are considered non-fatal to this theory due to the coda condition. These facts point toward an argument for syllable structure asymmetry in English.

In Stratal Harmonic Serialism, the syllabification pattern permitted at the stem level would be more restrictive than that found on the word level. This account is similar to the one proposed by Borowsky, given the similarities between lexical phonology and stratal frameworks.

4.2.6 Discussion

In the examples of syllable asymmetry shown here, there is a common involvement of morphology in the derivation of different syllabification patterns. While none of these cases are exactly like the cumulative patterns predicted the model in chapter 3 and discussed in section 4.1, they show a similar type of pattern in the different syllables generated at different levels and affix junctures. The resulting syllable inventories in these examples still fall into those predicted by frameworks like parallel OT, but the interactions at the morphology-phonology interface could potentially arise in natural language. Cases like these provide support for the typological predictions found in chapter 3.

4.3 Opacity

Opacity is a central issue in parallel OT, and many attempts to solve it include some derivational aspect. The original conception of opacity came from Kiparsky (1968, 1973) in the form of ordered rule interaction, with transparent feeding and bleeding rules and opaque counterfeeding and counterbleeding. While there has been work in classifying opaque interactions with a more fine-grained typology (Baković 2007), counterfeeding and counterbleeding have endured as crucial ways for classifying opaque interactions.

For the purposes of investigating opacity in Stratal Harmonic Serialism, I further classify opaque interactions by the locus of interaction, interlevel or intralevel. Both components of the framework have been shown to capture some aspects of opacity. Interlevel opacity is an interaction that could be captured by Stratal OT levels; that is, the interacting processes could be attributed to different levels of the morphology. Intralevel opacity is an interaction which would need to be accounted for within a single level of the grammar, either by the gradual derivation provided by Harmonic Serialism, or some other mechanism.

4.3.1 Interlevel

In this section, I provide examples of opaque interactions that are the result of processes occurring at different morphological levels. These cases have been used to support Stratal OT, but can similarly be applied to the Stratal component of Stratal Harmonic Serialism.

4.3.1.1 Counterbleeding

A case of counterbleeding opacity is one in which two interacting processes occur in separate levels; the later process would bleed the first, but because it applies in a different level, both processes apply. The result is a surface form in which the context of the first process in no longer present. A well known case of counterbleeding that fits this pattern is Canadian Raising (Joos 1942; Chomsky and Halle 1968), which is the interaction of stem-level[17] diphthong raising and a phrase-level flapping. Bermúdez-Otero (2003) provides an analysis of this opaque interaction in Stratal OT.

The first relevant process is diphthong raising, in which underlying /aɪ, ɑʊ/ → [əi, ʌʊ] before voiceless obstruents, as seen in (258). Crucially, diphthong raising is stem-level process, failing to apply across word boundaries, as in (258)b.

(258) Diphthong raising (Bermúdez-Otero 2003)

a.	nəif	'knife'	naɪvz	'knives'
	hʌʊs	'house'	hɑʊzɪz	'houses'
b.	ləifər	'lifer'	laɪ fər mi	'lie for me'

The second relevant process in flapping, in /t,d/ → [ɾ] after a stressed vowel, as seen in (259)a. Crucially, flapping is a phrase-level process, applying across word boundaries, as in (259)b.

(259) Flapping (Bermúdez-Otero 2003)

a.	fæt	'fat'	fæɾər	'fatter'
	mæd	'mad'	mæɾər	'madder'
b.	hɪt	'hit'	hi hɪɾ æn	'he hit Ann'
	hɪd	'hid'	hi hɪɾ æn	'he hid Ann'

Diphthong raising and flapping interact through voiced obstruents; diphthong raising applies in the context of voiceless, but not voiced, obstruents, and flapping converts the underlying voiceless /t/ into surface voiced [ɾ]. If flapping were to apply first, it would

[17] Bermúdez-Otero (2003) argues that this is a stem-level process due to evidence of diphthong not applying before word-level affixes like –ful, as in 'eyeful' [aɪfʊl] vs. 'eiffel' [əifəl], but for the purpose of this analysis the difference is not crucial, as stem- and word-level processes both precede phrase-level ones.

destroy the environment for raising to apply, thus bleeding diphthong raising; however, this is not the case, with raising applying before flapping, making this a counterbleeding interaction. The derivation for the opaque 'writing' as it contrasts with transparent 'riding' is shown in (260):

(260) Counterbleeding opacity in Canadian English

	'writing'	'riding'
UR	/raɪt-ɪŋ/	/raɪd-ɪŋ/
Raising	rəitɪŋ	raɪdɪŋ
Flapping	rəiɾɪŋ	raɪɾɪŋ
Surface	[rəiɾɪŋ]	[raɪɾɪŋ]

In the underlying representation, 'writing' and 'riding' differ only in the voicing of the alveolar stop. At the stem level, diphthong raising applies to 'writing,' where /aɪ/ →[əi] before voiceless /t/, but does not apply in 'riding' before voiced /d/. At the phrase level, the voicing difference between 'writing' and 'riding' is levelled, with flapping applying to both /t/ and /d/. Because flapping applied after raising, both are able to apply. This results in an opaque surface for 'writing' [rəiɾɪŋ], where the generalization of raised [əi] appearing before voiceless obstruents is not surface apparent.

To account for this change in Stratal OT, Bermúdez-Otero (2003) uses the markedness constraints ClearDiph, which assigns violations to the raised diphthongs [əi, ʌʊ], and ClipDiph, which assigns violations to [aɪ, ɑʊ] before voiceless obstruents. The markedness constraints used are Ident[mid] and Ident[low], each assigning violations for a change in vowel height. Bermúdez-Otero provides the stem-level ranking ClipDiph>>ClearDiph>>Ident[low], Ident[high] to permit raising for /raɪt-ɪŋ/ and phrase-level Ident[low]>>ClipDiph>>ClearDiph>>Ident[high], preventing any novel vowel raising, and remaining faithful to vowel height at the previous level, shown in tableaux (261) and (262):

(261) Stem level

/raɪt-ɪŋ/	ClipDiph	ClearDiph	Ident[low]	Ident[mid]
a. raɪtɪŋ	*!			
b. → rəitɪŋ		*	*	

(262) Phrase level

/rəit-ɪŋ/	Ident[low]	ClipDiph	ClearDiph	Ident[mid]
a. raɪtɪŋ			*	
b. → rəitɪŋ		*!		*

While the original ranking was proposed for Stratal OT, the presence of only a single operation of vowel raising at this level makes it easy to translate to Stratal Harmonic Serialism. An additional pass through the grammar would be required to reach convergence, which would happen in the next step due to the absence of any possible harmonically improving steps.

4.3.1.2 Counterfeeding

A case of counterfeeding opacity is one in which two interacting processes occur in separate levels; the later process would create the context for the earlier one to apply, but does not because it is ordered after. A case of counterfeeding that has been described in Stratal OT is the interaction between dorsal fricative assimilation and rhotic vocalization in German (Ito and Mester 2001).

The first relevant process is dorsal fricative assimilation, in which ç → x after back vowels, as shown in (263)a. Crucially, this is a stem-level process, due to the failure of application at morpheme boundaries, as shown in (263)b:

(263) Dorsal fricative assimilation (Ito and Mester 2001)

a. *Licht* lɪçt 'light' *Bucht* bʊxt 'bay'

Nächte nɛçtə 'night' *Nachte* nɑxt 'nights'

b. *Kuchen* kuːxən 'cake' *Kuh-chen* kuːçən 'little cow'

The second relevant process is rhotic vocalization, in which ʀ → ɐ̯ in coda position, as shown in (264) Ito and Mester (2001) note that this process is highly variable and not sensitive to morphology, suggesting that it is a post-lexical process.

(264) Rhotic vocalization (Ito and Mester 2001)

Türen tyːʀən 'doors' *Tur* tyːɐ̯ 'door'

Ohren oːʀən 'ears' *Ohr* oːɐ̯ 'ear'

This processes interact opaquely when rhotic vocalization creates a back vowel that should feed the assimilation process, but due to the ordering does not, resulting in a counterfeeding interaction. Consider the opaque case of 'durch' as compared to the transparent 'buch' and 'tur,' as shown in (265):

(265) Counterfeeding opacity in German

	durch	buch	tur
UR	/dʊʀç/	/buːç/	/tyːʀ/
Assimilation	dʊʀç	buːx	tyːʀ
Vocalization	dʊɐ̯ç	buːx	tyːɐ̯
Surface	[dʊɐ̯ç]	[buːx]	[tyːɐ̯]

Because the /ç/ in 'durch' is not immediately preceded by a back vowel, assimilation does not apply, as it does in 'buch.' In the vocalization step, the /ʀ/ is vocalized to [ɐ̯], a back vowel. The surface form of 'durch' contains the sequence [ɐ̯ç], where the generalization that the [x] allomorph appears after back vowels is not surface true.

Because there is evidence that the assimilation and vocalization processes occur at different levels of the grammar, Ito and Mester (2001) provide a Stratal OT analysis of this interaction, where assimilation applies at the lexical level and vocalization applies at the post-lexical level.

For the lexical level, they provide the ranking VEL>>*x>>Id(back), where VEL is a markedness constraint assigning a violation to an occurrence of ç followed by a back vowel. This ranking allows assimilation for cases like 'buch,' as in (266):

(266) Lexical level: assimilation (Ito and Mester 2001)

/buːç/	VEL	*x	Id(back)
a. buːç	*!		
b. → buːx		*	*

Additionally, vocalization does not apply at the lexical level, which is achieved through the additional constraints, as shown in (267).

(267) Lexical level: assimilation (Ito and Mester 2001)

/dʊʀç/	*ʀ̯	*Coda/ʀ	*ʀ	Id(cons)	VEL	*x	*ç	Id(back)
a. → dʊʀç		*	*				*	
b. dʊʀx		*	*			*!		*
c. dʊʀ̯ç	*!			*	*		*	
d. dʊʀ̯x[18]	*!			*		*		*

At the postlexical level, the constraints are reranked. While the process of vocalization did not apply at the lexical level, it applies at the postlexical level with the ranking *Coda/ʀ>>* ʀ̯>> Id(cons), as in (268):

(268) Postlexical level: vocalization (Ito and Mester 2001)

/tyːʀ/	*Coda/ʀ	* ʀ̯	Id(cons)
a. tyːʀ	*!		
b. → tyːʀ̯		*	*

Adding in the remaining constraints, the reranking of those constraints means that assimilation no longer applies at the postlexical level, as in (269):

[18] While this candidate is possible in Stratal OT, it would not be possible in Stratal Harmonic Serialism because it undergoes two operations in one step: assimilation and vocalization.

(269) Postlexical level: vocalization (Ito and Mester 2001)

/dʊʀç/	*Coda/ʀ	*ɐ̯	*ʀ	Id(cons)	Id(back)	VEL	*x	*ç
a. dʊʀç	*!		*					*
b. dʊʀx	*!		*		*		*	
c. → dʊɐ̯ç		*		*		*		*
d. dʊɐ̯x[19]		*		*	*!		*	

While this analysis was proposed for Stratal OT, the selected candidates only undergo one operation each, so the outcome is nearly the same; each step would still require another pass through the grammar until convergence, but this would occur in the next step as there are no remaining harmonically improving operations.

4.3.2 Intralevel

In this section, I discuss cases of opaque interactions that would need to be accounted for in a single level.

4.3.2.1 Counterbleeding

A case of counterbleeding that occurs within a single level is the interaction between nasal place assimilation and cluster simplification in Catalan (Mascaró 1976; Kiparsky 1985). While it has been argued that this interaction cannot be accounted for in Stratal OT (McCarthy 2007a), it is possible in Stratal Harmonic Serialism, as I will demonstrate here.

---

[19] Same as above.

The first relevant process is nasal assimilation, in which /n/ assimilates in place to a following consonant, as in (270). Assimilation is not as productive for the other phonemic nasals: /m/ only alternates with [ɱ] (270) and /ŋ/ and /ɲ/ do not assimilate in any context (270).

(270) Nasal Assimilation (Mascaró 1976; Kiparsky 1985)

a.	so[n] amics	'they are friends'
	so[m] pocs	'they are few'
	so[ɱ] feliços	'they are happy'
	so[ŋ] grans	'they are big'
b.	so[m] amics	'we are friends'
	so[m] pocs	'we are few'
	so[ɱ] feliços	'we are happy'
c.	ti[ŋ] pa	'I have bread'
	a[ɲ] feliç	'happy year'

The second relevant process is cluster simplification, where the second consonant in a sequence of two homorganic stops deletes word-finally or before a consonant. In (271)a, the final consonant in /kamp/ deletes in all contexts at the word boundary. In (271)b, the /p/ is retained only before a vowel-initial suffix.

(271) Cluster simplification

a.	kam es	‘the field is’
	kam sigi	‘the field were (subj.)’
	kam	‘the field’
b.	kams	‘fields’
	kampɛt	‘little field’

These two processes interact in forms like /bɛnk/ → [bɛŋ], as shown in (272).

(272) Counterbleeding in Catalan

	venc ‘I sell’
Underlying	/bɛnk/
Place assimilation	bɛŋk
Cluster simplification	bɛŋ
Surface	[bɛŋ]

The process of place assimilation applies first, resulting in a surface velar nasal, followed by cluster simplification. If cluster simplification applied first, it would bleed nasal assimilation by deleting the conditioning environment, making this counterbleeding opacity. Cluster simplification applies within words, but not across word boundaries, so it cannot apply at the post-lexical level. Unlike the cases of interlevel opacity, this is a case of intralevel opacity that cannot be resolved through stratal levels.

This is a traditional case of opacity that Parallel OT (or a single level of Stratal OT) cannot account for because the desired winner is harmonically bounded by the transparent candidate, as in (273).

(273) Desired winner loses in Parallel OT

/bɛnk/	Agree[Place]	*Complex	Max	Id[Place]
a. bɛnk	W1	W1		
b. 💣bɛn			1	
c. → bɛŋ			1	1
d. bɛŋk		W1		1

The desired winner (c) and the unintended winner (b) both have violations of Max, but (c) also has an additional violation of Id[Place], so it cannot win under any possible ranking of the constraints.

In Harmonic Serialism, the issue of opacity is not resolved under a theory of operations in which deletion occurs in a single step. For this analysis, we require at least two operations: deletion and place assimilation. For deletion, assuming this is a one-step process, the necessary constraint ranking is *Complex>>Max, to allow deletion of a consonant in a complex cluster. For assimilation, we require a markedness constraint that dominates the faithfulness constraint Id[Place]. Here I use the Share[Place] markedness constraint in place of Agree[Place], as is preferred for Harmonic Serialism, assigning violations to adjacent segments that do not share a place feature (McCarthy 2009), giving the ranking Share[Place]>>Id[Place]. With the input candidate /bɛnk/, the first step of the derivation is shown in (274).

(274) Step 1: Deletion

/bɛnk/	Share[Place]	*Complex	Max	Id[Place]
a. bɛnk	W1	W1		
b. 💣bɛn			1	
c. → bɛŋk		W1		1

In step 1, the desired candidate (c) undergoes the place assimilation operation, incurring a violation of Id[Place] and losing a violation of Share[Place]. However, there is also a fell-swoop candidate (b), which repairs violatons of Share[Place] and *Complex with a single step of deletion. This derivation converges in step 2:

(275) Step 2: Converge

/bɛn/	Share[Place]	*Complex	Max	Id[Place]
a. 💣bɛn				
b. →bɛŋ				1

After the incorrect intermediate form selected in step 1, the desired candidate cannot win in step 2, due to the deletion of the conditioning environment that would motivate nasal place assimilation. In the absence of a faithful candidate with a Share[Place] violation, a violation of Id[Place] is not harmonically improving. While the correct form cannot be derived under this analysis, Harmonic Serialism is not only a theory of constraint interaction, but also one of operations.

The use of gradual deletion in Harmonic Serialism has already been shown to be useful as a way to solve the Too Many Solutions problem in cluster reduction (McCarthy 2008). While there is a typological preference for deletion of C_1 in intervocalic C_1C_2

clusters, Parallel OT permits the mapping of both /patka/ → [paka] and /patka/ → [pata]. Under this analysis, deletion occurs as two separate operations: deletion of a place feature and deletion of a segment slot. Place feature deletion is motivated by licensing of codas, due to the CodaCond constraint. Under gradual deletion in Harmonic Serialism, the derivation /patka/ → paHka → [paka] is preferred, aligning with the typological preference.

The use of gradual deletion in cluster reduction can be extended to counterbleeding opacity in Catalan. With deletion as a two step process, the fell-swoop candidate fatal to the previous analysis is no longer possible under Gen; that is, the single operation of place feature deletion cannot repair violations of both *Share[Place] and *Complex.

For this analysis, the faithfulness constraint Max has been split into Max[Place] and Max[Seg], to govern the two operations of place feature deletion and segment slot deletion. I also include the ParseSeg constraint to include syllable parsing in the set of operations, which occur in the first two steps of the derivation. Following previous analyses, core syllabification and coda adjunction are possible operations. In the first step a CV syllable is parsed, shown in (276):

(276) Step 1: Core syllabification

/bɛnk/	*Complex	ParseSeg	Share[Place]	Id[Place]	Max[Place]	Max[Seg]
a. bɛnk		W4	1			
b. →(bɛ)nk		2	1			
c. b(ɛ)nk		W3	1			
d. (b)ɛnk		W3	1			

In the first step, a CV syllable is parsed, removing two violations of the Parse Seg constraint. In the following step, shown in (277), the following consonant is parsed into the coda position, removing an additional violation of ParseSeg.

(277) Step 2: Coda adjunction

/(bɛ)nk/	*Complex	ParseSeg	Share[Place]	Id[Place]	Max[Place]	Max[Seg]
a. (bɛ)nk		W2	1			
b. →(bɛn)k		1	1			

In the next step, the opaque interactions come into play. There are four relevant candidates permitted by Gen. Candidate (a) is the faithful candidate, with remaining violations of ParseSeg and Share[Place]. Candidate (b) undergoes place assimilation, improving Share[Place] at the expense of Id[Place], and maintaining a violation of ParseSeg. Candidate (c) undergoes complex coda adjunction, incurring a fatal violation of *Complex at the expense of ParseSeg, so this candidate is not actually harmonically improving. Candidate (d) undergoes place feature deletion at the expense of Max[Place]. Crucially, there is no fell-swoop candidate due to the lack of a complete deletion operation, so candidate (b) is the most harmonically improving.

(278) Step 3: Place agreement

/(bɛn)k/	*Complex	ParseSeg	Share[Place]	Id[Place]	Max[Place]	Max[Seg]
a. (bɛn)k		1	W1	L		
b. → [dor] / \ (bɛŋ)k		1		1		
c. (bɛnk)	W1	L	W1	L		
d. (bɛn)H		1	W1	L	W1	

In the next step, cluster reduction can occur as the result of segment slot deletion, shown by candidate (b) in (279).

(279) Step 4: Delete segment slot

[dor] / \ /(bɛŋ)k/	*Complex	ParseSeg	Share[Place]	Id[Place]	Max[Place]	Max[Seg]
a. [dor] / \ (bɛŋ)k		W1				L
b. → [dor] | (bɛŋ)						1
c. [dor] /\ (bɛŋk)	W1					L
d. [dor] | (bɛŋ)H		W1	W1	W1		L

With no remaining harmonically improving steps, the derivation will converge in the next step, shown in (280):

(280) Step 5: Converge

[dor] | /(bɛŋ)/	*Complex	ParseSeg	Share[Place]	Id[Place]	Max[Place]	Max[Seg]
a. [dor] | →(bɛŋ)						
b. (bɛn)					W1	

From this analysis, I show that Harmonic Serialism offers some mechanisms for handling counterbleeding opacity within a single level. In this case, gradual derivation can be used to restrict the more globally optimal fell-swoop candidates in favor of locally-optimal opaque candidates, when those operations are possible under a ranking of the constraints.

4.3.2.2 Counterfeeding

While gradual derivation may be a successful approach for handling counterbleeding opacity, this is not the case for counterfeeding due to the different nature of the resulting surface forms of counterfeeding interactions. While counterbleeding opacity results in surface forms that are not transparent due to the loss of the conditioning environment, they can potentially be handled in a constraint-based framework if the set of operations can be ordered in a meaningful way, as was demonstrated in the previous section. On the other hand, counterfeeding opacity has contexts where a process can apply but does not. Within traditional constraint interaction, there is no way to block the selection of a candidate, which would otherwise be permitted, based solely on its derivational history. One potential use of Harmonic Serialism gradualness to block potential candidates in a single level is through the building of structure between processes, though this would require concrete assumptions about the nature of structure build beyond the syllable.

4.3.3 Discussion

In this section, I have discussed the current state of opacity in Stratal Harmonic Serialism. Interlevel opacity is fairly straightforward for the framework, as the stratal component permits reranking of constraints between levels, thus allowing constraints governing interacting processes to be ordered at different levels. For intralevel opacity, on the other hand, Stratal Harmonic Serialism does not offer a general solution; it inherits the issues faced by Harmonic Serialism and to some extent, parallel OT. However, I have offered a novel analysis of some types of interlevel counterbleeding interactions through the gradualness of Harmonic Serialism levels, here showing the case of Catalan. There is remaining work to be done on offering approaches to other cases of intralevel opacity, namely counterfeeding interactions, as well as other cases of counterbleeding that may not be covered by this approach.

Chapter 5 Conclusion

In this chapter, I provide a summary of the dissertation and discuss remaining issues.

5.1 Summary of the dissertation

This dissertation has argued for a framework, Stratal Harmonic Serialism, as a way to address some of the typological issues of undergeneration and overgeneration within the strictly parallel constraint-based framework of classic Optimality Theory.

One of the major issues for OT is the problem of opacity, a case of undergeneration which, as a fully parallel system, OT has trouble addressing without some form of serialism introduced into the model. I address the issue of opacity in two ways. In chapter 2, I provide a case study of tonal opacity in Kikerewe. The phonology of Kikerewe has a number of complex processes that interact locally. As I show, the derivational mechanisms of Stratal Harmonic Serialism are well suited to account for the tonology of Kikerewe. In chapter 4, I provide the status of Stratal Harmonic Serialism within the typology of opaque interactions. As I show, Stratal Harmonic Serialism can account for interlevel counterfeeding and counterbleeding opacity, due to the stratal component that permits reranking between levels. Additionally, the Harmonic Serialism component can account for some cases of intralevel opacity, due to the gradual component, as I show with a case of counterbleeding in Catalan. While Stratal Harmonic Serialism does not provide a general solution to opacity, that is not necessarily fatal to the theory, but the remaining cases of opaque interactions would still need to be accounted for, possibly by some other mechanism.

Another major issue for OT is the overgeneration of some unattested patterns, due to the preference for globally optimal candidates. As addressed in the Harmonic

Serialism literature, gradual derivation generates locally optimal candidates, which have been shown to be typologically preferred. While Harmonic Serialism is a restrictive framework, there is a possibility that the stratal component of Stratal Harmonic Serialism could overgenerate unattested grammars. In chapter 3, I provide a factorial typology of syllable structure in Stratal Harmonic Serialism. While this typology does generate some additional syllable structure typologies beyond those predicted by a fully parallel OT, they do not violate implicational markedness. In chapter 4, I discuss some of the implications of this typology with regard to cumulativity and asymmetry, and some attested examples from natural language.

5.2 Remaining issues

One the of the remaining issues is the status of operations and representations within the syllable structure typology. While the current analysis has shown some promising results, there are many additional factors that could play a role, but beyond a certain number of constraints, a factorial typology becomes computationally intractable. Some of the remaining issues include the status of minor syllables and a theory of epenthesis. As I have shown, the inclusion of minor syllables into the typology produces some undesired results. Given that they are a key assumption as an intermediate step for epenthesis within gradual syllabification, this issue would need to be resolved with an adaption to the set constraints, operations, or representations. Another issue central to syllabic theory is sonority, which has not been addressed. The implications of adding constraints governing sonority in syllabification in Stratal Harmonic Serialism remains an area for future study.

Another remaining issue is the question of opacity. While this framework makes some progress in its range of coverage of opaque interactions, it has certainly not provided a complete account. Leaving some interactions to other mechanisms requires investigation into the topic of how those can be incorporated into Stratal Harmonic Serialism to provide more comprehensive coverage of opaque phenomena.

References

Abu-Salim, Issam M. 1982. "A Reanalysis of Some Aspects of Arabic Phonology: A Metrical Approach." Doctoral dissertation, Urbana-Champaign, IL: University of Illinois at Urbana-Champaign.

Albright, Adam, Giorgio Magri, and Jennifer Michaels. 2008. "Modeling Doubly Marked Lags with a Split Additive Model." In *Proceedings of the 32nd Annual Boston University Conference on Lanugage Development*, 1:36–47.

Baković, Eric. 1999. "Assimilation to the Unmarked." In *University of Pennsylvania Working Papers in Linguistics: Proceedings of the 24th Annual Penn Linguistics Colloquium,* edited by Jim Alexander, Na-Rae Han, and Michelle Fox. Philadelphia: Department of Linguistics, University of Pennsylvania.

Baković, Eric. 2007. "A Revised Typology of Opaque Generalisations." *Phonology* 24 (2): 217–59.

Baković, Eric. 2011. "Opacity Deconstructed." In *The Blackwell Companion to Phonology.*, edited by Marc van Oostendorp, Colin J. Ewen, Elizabeth Hume, and Keren Rice. Oxford: Blackwell.

Bane, Max, and Jason Riggle. 2012. "Consequences of Candidate Omission." *Linguistic Inquiry* 43 (4): 695–706.

Beckman, Jill. 1997. "Positional Faithfulness, Positional Neutralisation and Shona Vowel Harmony." *Phonology* 14 (1): 1–46.

Beckman, Jill. 1998. "Positional Faithfulness." Amherst: University of Massachusetts Amherst.

Bennett, Ryan. 2016. "Mayan Phonology." *Language and Linguistics Compass* 10 (10): 469–514.

Benua, Laura. 1995. "Identity Effects in Morphological Truncation." *University of Massachusetts Occasional Papers in Linguistics* 18: 77–136.

Benua, Laura. 1997. "Transderivational Identity: Phonological Relations between Words." Doctoral dissertation, Amherst, MA: University of Massachusetts Amherst.

Bermúdez-Otero, Ricardo. 1999. "Constraint Interaction in Language Change: Quantity in English and Germanic." Doctoral dissertation, Manchester, UK: University of Manchester.

Bermúdez-Otero, Ricardo. 2003. "The Acquisition of Phonological Opacity." In *Variation within Optimality Theory: Proceedings of the Stockholm Workshop on Variation within Optimality Theory*, 25–36.

Bickmore, Lee. 1996. "Bantu Tone Spreading and Displacement as Alignment and Minimal Misalignment." *Albany, NY: University at Albany. [Available on Rutgers Optimality Archive.]*.

Bickmore, Lee. 1999. "High Tone Spread in Ekegusii Revisited: An Optimality Theoretic Account." *Lingua* 109 (2): 109–53.

Black, H. Andrew. 1993. "Constraint-Ranked Derivation: A Serial Approach to Optimization." Doctoral dissertation, Santa Cruz, CA: University of California, Santa Cruz.

Blevins, Juliette. 1995. "The Syllable in Phonological Theory." In *The Handbook of Phonological Theory*, edited by John Goldsmith, 206–44. Carnbridge, MS: Blackwell.

Blumenfeld, Lev A. 2006. "Constraints on Phonological Interactions." Doctoral dissertation, Stanford, CA: Stanford University.

Booij, Geert. 1996. "Lexical Phonology and the Derivational Residue." In *Current Trends in Phonology: Models and Methods*, edited by Jacques Durand and Bernard Laks, 69–96. Salford: University of Salford.

Booij, Geert. 1997. "Non-Derivational Phonology Meets Lexical Phonology." In *Derivations and Constraints in Phonology*, edited by Iggy Roca, 261–88. Oxford: Oxford University Press.

Borowsky, Toni. 1989. "Structure Preservation and the Syllable Coda in English." *Natural Language & Linguistic Theory* 7 (2): 145–166.

Cardinaletti, Anna, and Lori Repetti. 2009. "Phrase-Level and Word-Level Syllables: Resyllabification and Prosodization of Clitics." In *Phonological Domains: Universals and Deviations*, edited by Janet Grijzenhout and Baris Kabak, 16:79–104. Berlin: Mouton de Gruyter.

Cassimjee, Farida, and Charles Kisseberth. 1998. "Optimal Domains Theory and Bantu Tonology: A Case Study from Isixhosa and Shingasidja." In *Theoretical Aspects of Bantu Tone*, edited by Larry Hyman and Charles Kisseberth, 33–132. Stanford, CA: CSLI.

Chomsky, Noam, and Morris Halle. 1968. *The Sound Pattern of English.* New York: Harper and Row.

Clements, George N., and Samuel Jay Keyser. 1983. *CV Phonology: A Generative Theory of the Syllable*. Cambridge, MA: MIT Press.

Collie, Sarah. 2008. "English Stress Preservation and Stratal Optimality Theory." Doctoral dissertation, Edinburgh: University of Edinburgh.

Dayley, Jon P. 1985. *Tzutujil Grammar (University of California Publications in Linguistics)*. Berkeley: University of California Press.

Dell, François, and Mohamed Elmedlaoui. 1985. "Syllabic Consonants and Syllabification in Imdlawn Tashlhiyt Berber." *Journal of African Languages and Linguistics* 7 (2): 105–130.

Downing, Laura J. 1998. "On the Prosodic Misalignment of Onsetless Syllables." *Natural Language & Linguistic Theory* 16 (1): 1–52.

Downing, Laura J. 1999. "Prosodic Stem ≠ Prosodic Word in Bantu." In *Studies on the Phonological Word*, edited by T. Allan Hall and Ursala Kleinhenz, 174:73–98. John Benjamins.

Elfner, Emily. 2009. "Syllabification and Stress-Epenthesis Interactions in Harmonic Serialism." University of Massachusetts Amherst.

Face, Timothy. 2002. "Re-Examining Spanish 'Resyllabification.'" In *Current Issues in Romance Languages: Selected Papers from the 29th Linguistic Symposium on Romance Languages (LSRL), Ann Arbor, 8 11 April 1999*, 220:81. John Benjamins Publishing.

Farwaneh, Samira. 1995. "Directionality Effects in Arabic Dialect Syllable Structure." Doctoral dissertation, Salt Lake City, UT: University of Utah.

Fitzgerald, Colleen M. 2002. "Tohono O'odham Stress in a Single Ranking." *Phonology* 19: 253–271.

Goldsmith, John. 1976. "Autosegmental Phonology." Doctoral dissertation, Cambridge, MA: MIT.

Goldsmith, John. 1984. "Meeussen's Rule." *Language Sound Structure*, 245–259.

Goldsmith, John. 1990. *Autosegmental and Metrical Phonology*. Vol. 1. Basil Blackwell.

Greenberg, Joseph. 1966. Synchronic and diachronic universals on phonology. *Language* 42: 508-17.

Idsardi, William J. 2000. “Clarifying Opacity.” *The Linguistic Review* 17 (2–4): 337–350.

Ito, Junko. 1986. “Syllable Theory in Prosodic Phonology.” Doctoral dissertation, Amherst, MA: UMass Amherst.

Ito, Junko. 1989. “A Prosodic Theory of Epenthesis - Google Scholar.” *Natural Language & Linguistic Theory* 7: 217–60.

Ito, Junko, and Armin Mester. 1996. “Rendaku I: Constraint Conjunction and the OCP.” *Ms. University of California, Santa Cruz*.

Ito, Junko, and Armin Mester. 2001. “Structure Preservation and Stratal Opacity in German.” *Segmental Phonology in Optimality Theory*, 261–295.

Jakobson, Roman. 1963. “Implications of Language Universals for Linguistics.” In *Universals of Language*, edited by Joseph Greenberg, 263–278. Cambridge, MA: MIT Press.

Jarosz, Gaja. 2010. “Implicational Markedness and Frequency in Constraint-Based Computational Models of Phonological Learning.” *Journal of Child Language* 37 (3): 565–606.

Jesney, Karen. 2011. “Positional Faithfulness, Non-Locality, and the Harmonic Serialism Solution.” In *Proceedings of the North East Linguistic Society*. Vol. 39.

Joos, Martin. 1942. “A Phonological Dilemma in Canadian English.” *Language*, 141–144.

Kaisse, Ellen M. 1997. “Aspiration and Resyllabification in Argentinian Spanish.” *University of Washington Working Papers in Linguistics* 15: 199–209.

Kaisse, Ellen M. 1998. “Resyllabification: Evidence from Argentinian Spanish.” In *Formal Perspectives on Romance Linguistics*, edited by J. Marc Authier, Barbara E. Bullock, and Lisa A. Reed, 197–210. Philadelphia, PA: John Benjamins.

Karttunen, Lauri. 1993. “Finite-State Constraints.” In *The Last Phonological Rule*, edited by John Goldsmith, 173–194. Chicago: University of Chicago Press.

Karttunen, Lauri. 2006. “The Insufficiency of Paper-and-Pencil Linguistics: The Case of Finnish Prosody.” *Intelligent Linguistic Architectures: Variations on Themes by Ronald M. Kaplan*, 287–300.

Kimper, Wendell. 2008. "Local Optionality and Harmonic Serialism." *Manuscript. Amherst, MA: University of Massachusetts Amherst.[Available on Rutgers Optimality Archive, ROA-988.]*.

Kimper, Wendell. 2011. "Locality and Globality in Phonological Variation." *Natural Language & Linguistic Theory* 29 (2): 423–465.

Kiparsky, Paul. 1968. "Linguistic Universals and Linguistic Change." In *Universals in Linguistic Theory*, edited by Emmon Bach and Robert Harms, 170–202. New York: Holt, Rinehart, and Winston.

Kiparsky, Paul. 1973. *Abstractness, Opacity and Global Rules*. Bloomington: Indiana University Linguistics Club.

Kiparsky, Paul. 1982. "Lexical Phonology and Morphology." In *Linguistics in the Morning Calm*, edited by I.S. Yang, 3–91. Seoul: Hanshin.

Kiparsky, Paul. 1985. "Some Consequences of Lexical Phonology." *Phonology* 2: 85–138.

Kiparsky, Paul. 2000. "Opacity and Cyclicity." *The Linguistic Review* 17 (2–4): 351–366.

Koskenniemi, Kimmo. 1983. "Two-Level Morphology." *A General Computational Model for Word-Form Recognition and Production. Department of General Linguistics. University of Helsinki, Finland*.

Leben, William Ronald. 1973. "Suprasegmental Phonology." Doctoral dissertation, Cambridge, MA: MIT.

Levelt, Clara C., Niels O. Schiller, and Willem J. Levelt. 2000. "The Acquisition of Syllable Types." *Language Acquisition* 8 (3): 237–64.

Lin, Yen-Hwei. 1997. "Cyclic and Noncyclic Affixation in Piro." In *G.E. Booij & J.M. van de Weijer (Eds.) Phonology in Progress - Progress in Phonology*, 167–88. The Hague: Holland Academic Graphics.

Lombardi, Linda. 1999. "Positional Faithfulness and Voicing Assimilation in Optimality Theory." *Natural Language & Linguistic Theory* 17 (2): 267–302.

Lombardi, Linda. 2001. "Why Place and Voice Are Different: Constraint-Specific Alternations in Optimality Theory." *Segmental Phonology in Optimality Theory: Constraints and Representations*, 13–45.

Lubowicz, Anna. 2002. "Derived Environment Effects in Optimality Theory." *Lingua* 112 (4): 243–280.

Mascaró, Joan. 1976. "Catalan Phonology and the Phonological Cycle." Doctoral dissertation, Cambridge, MA: MIT.

Matteson, Esther. 1965. *The Piro (Arawakan) Language*. Berkeley: University of California Press.

McCarthy, John J. 1979. "Formal Problems in Semitic Morphology and Phonology." Doctoral dissertation, Cambridge, MA: MIT.

McCarthy, John J. 1996. "Remarks on Phonological Opacity in Optimality Theory." In *Studies in Afroasiatic Grammar: Papers from the Second Conference on Afroasiatic Linguistics*, edited by Jacqueline Lecarme, Jean Lowenstamm, and Ur Shlonsky, 215–43. The Hague: Holland Academic Graphics.

McCarthy, John J. 1999. "Sympathy and Phonological Opacity." *Phonology* 16 (3): 331–99.

McCarthy, John J. 2000. "Harmonic Serialism and Parallelism." In *Proceedings of the North East Linguistic Society*, edited by Masako Hirotani, 30:501–24. Amherst, MA: GLSA Publications.

McCarthy, John J. 2007a. *Hidden Generalizations: Phonological Opacity in Optimality Theory*. London: Equinox.

McCarthy, John J. 2007b. "Restraint of Analysis." In *Freedom of Analysis*, edited by Sylvia Blaho, Patrik Bye, and Martin Kraemer, 203–31. The Hague: Mouton.

McCarthy, John J. 2008. "The Gradual Path to Cluster Simplification." *Phonology* 25 (2): 271–319.

McCarthy, John J. 2009. "Harmony in Harmonic Serialism." Manuscript. University of Massachusetts, Amherst.

McCarthy, John J., and Abigail Cohn. 1998. "Alignment and Parallelism in Indonesian Phonology." *Working Papers of the Cornell Phonetics Laboratory* 12.6.

McCarthy, John J., and Alan Prince. 1993a. "Generalized Alignment." In *Yearbook of Morphology 1993*, 79–153. Springer.

McCarthy, John J., and Alan Prince. 1993b. "Prosodic Morphology: Constraint Interaction and Satisfaction." New Brunswick, NJ: Rutgers University Center for Cognitive Science.

Meeussen, Achille E. 1963. "Morphotonology of the Tonga Verb." *Journal of African Languages* 2 (Part 1).

Mohanan, Karuvannur P. 1982. "Lexical Phonology." Doctoral dissertation, Cambridge, MA: MIT.

Mohanan, Karuvannur P. 1986. *The Theory of Lexical Phonology*. Reidel.

Morén, Bruce. 2000. "The Puzzle of Kashmiri Stress: Implications for Weight Theory." *Phonology* 17 (3): 365–396.

Moreton, Elliot, and Paul Smolensky. 2002. "Typological Consequences of Local Constraint Conjunction." https://rucore.libraries.rutgers.edu/rutgers-lib/46746/.

Morris, Richard E. 2002. "Coda Obstruents and Local Constraint Conjunction in North-Central Peninsular Spanish." In *Current Issues in Linguistic Theory: Selected Papers from the 29th Linguistic Sympo- Sium on Romance Languages*, edited by Diana Cresti, Teresa Satterfield, and Christina Tortora. Amsterdam: Benjamins.

Myers, Scott. 1994. "OCP Effects in Optimality Theory." Manuscript. University of Texas, Austin.

Myers, Scott. 1997. "OCP Effects In Optimality Theory." *Natural Language and Linguistic Theory* 15 (8474892).

Odden, David. 1995. "The Status of Onsetless Syllables in Kikerewe." *Working Papers in Linguistics-Ohio State University Department of Linguistics*, 89–110.

Odden, David. 2000. "Opacity and Ordering: H-Deletion in Kikerewe." *The Linguistic Review* 17 (2–4): 323–336.

Odden, David. 2008. "Ordering." In *Rules, Constraints, and Phonological Phenomena*, edited by Bert Vaux and Andrew Nevins, 61–120. Oxford: Oxford University Press.

Onn, Farid Mohd. 1976. "Aspects of Malay Phonology and Morphology." Doctoral dissertation, Urbana-Champaign, IL: University of Illinois at Urbana-Champaign.

Padgett, Jaye. 2002. *Constraint Conjunction versus Grounded Constraint Subhierarchies in Optimality Theory*.

Pater, Joe. 1999. "Austronesian Nasal Substitution and Other NC Effects." In *The Prosody-Morphology Interface*, edited by René Kager, Harry van der Hulst, and Wim Zonneveld, 310–343. Cambridge: Cambridge University Press.

Pater, Joe. 2009. "Weighted Constraints in Generative Linguistics." *Cognitive Science* 33 (6): 999–1035.

Prince, Alan, and Paul Smolensky. 1993. "Optimality Theory: Constraint Interaction in Generative Grammar." [Available on Rutgers Optimality Archive, ROA-537].

Pruitt, Kathryn. 2010. "Serialism and Locality in Constraint-Based Metrical Parsing." *Phonology* 27 (3): 481–526.

Pruitt, Kathryn. 2012. "Stress in Harmonic Serialism." Doctoral dissertation, Amherst, MA: University of Massachusetts Amherst.

Rubach, Jerzy. 2000. "Glide and Glottal Stop Insertion in Slavic Languages: A DOT Analysis." *Linguistic Inquiry* 31 (2): 271–317.

Rubach, Jerzy. 2003a. "Duke-of-York Derivations in Polish." *Linguistic Inquiry* 34 (4): 601–629.

Rubach, Jerzy. 2003b. "Polish Palatalization in Derivational Optimality Theory." *Lingua* 113 (3): 197–237.

Shaw, Jason. 2009. "Compensatory Lengthening via Mora Preservation in OT-CC: Theory and Predictions." https://rucore.libraries.rutgers.edu/rutgers-lib/41015/.

Smolensky, Paul. 1995. "On the Internal Structure of the Constraint Component Con of UG." Handout of talk presented at UCLA. Los Angeles, CA.

Smolensky, Paul. 2006. "Optimality in Phonology II: Harmonic Completeness, Local Constraint Conjunction, and Feature Domain Markedness." *The Harmonic Mind: From Neural Computation to Optimality-Theoretic Grammar* 2: 27–160.

Steriade, Donca. 1982. "Greek Prosodies and the Nature of Syllabification." Doctoral dissertation, MIT: Cambridge, MA.

Steriade, Donca. 2001. "The Phonology of Perceptibility Effects: The P-Map and Its Consequences for Constraint Organization." Manuscript. UCLA.

Wilson, Colin. 2000. "Targeted Constraints: An Approach to Contextual Neutralization in Optimality Theory." Doctoral dissertation, Baltimore, MD: Johns Hopkins University.

Wolf, Matthew. 2012. "Inversion of Stress-Conditioned Phonology in Stratal OT." Manuscript. Yale University.

Yip, Moira. 2002. *Tone*. Cambridge: Cambridge University Press.

Zimmermann, Eva. 2013. “Vowel Deletion as Mora Usurpation: The Case of Yine.” *Phonology* 30 (1): 125–163.

Appendix

In section appendix, I provide key excerpts of the code used to generate the typological predictions in chapter 3.

```
def Eval(inp,ranking):
    tab = get_tableau(inp,ranking)
    cands = Gen(inp)
    winners = range(0,len(cands))
    for i in range(len(tab)):
        mins = min(tab[i])
        maxs = max(tab[i])
        if mins != maxs:
            temps = []
            vals = []
            for j in range(len(tab[i])):
                if tab[i][j]> mins:
                    if j in winners:
                        temps.append(j)
                        vals.append(tab[i][j])
            if winners != temps:
                for temp in temps:
                    if temp in winners:
                        winners.remove(temp)
            if winners == temps:
                min2 = min(vals)
                max2 = max(vals)
                for temp in temps:
                    if tab[i][temp] > min2:
                        winners.remove(temp)
    winner = cands[winners[0]][0]
    return winner

def Eval_loop(inp,ranking):
    winner = Eval(inp,ranking)
    while inp != winner:
        inp,winner = winner,Eval(winner,ranking)
    return winner


def get_all_outputs(inps,ranking):
    outs = []
    for inp in inps:
        output = Eval_loop(inp,ranking)
```

```
            outs.append([inp,output])
        return outs

    def get_all_rankings(cons):
        rankings = itertools.permutations(cons, len(cons))
        rank_list = [list(rank) for rank in rankings]
        return rank_list

    def get_inventory(mapping):
        inventory = []
        for pair in mapping:
            syls = pair[1].split('_')
            for syl in syls:
                if syl.islower():
                    newsyl = '_'+syl+'_'
                    if newsyl not in inventory:
                        inventory.append(newsyl)
        return inventory

    def get_inventory_syls(mapping):
        inventory = []
        for pair in mapping:
            syls = pair[1].split('_')
            newsyl=''
            for syl in syls:
                if syl.islower():
                    newsyl = newsyl+'_'+syl+'_'
                    if newsyl not in inventory:
                        inventory.append(newsyl)
        return inventory


    def get_L1_mappings(inps,cons):
        rankings = get_all_rankings(cons)
        mappings = []
        inventories = []
        inputs = []
        n = 1
        for rank in rankings:
            output = get_all_outputs(inps, rank)
            if output not in mappings:
                inven = sorted(get_inventory(output))
                inven_syls = sorted(get_inventory_syls(output))
                print 'Ranking #', n, rank
                for pair in output:
                    print 'Input: ', pair[0], '   Output: ', pair[1]
```

```
            print inven
            mappings.append(output)
            if inven not in inventories:
inventories.append(inven)
            if inven_syls not in inputs:
inputs.append(inven_syls)
            n+=1
    return mappings, inventories, inputs


def get_L2_mappings(inputs,cons):
    rankings = get_all_rankings(cons)
    inventories = []
    n= 1
    for inps in inputs:
        print 'L1 Input: ', inps
        temp = []
        for rank in rankings:
            output = get_all_outputs(inps, rank)
            inven = sorted(get_inventory(output))
            if inven not in inventories:
inventories.append(inven)
            if inven not in temp:
                print 'Ranking #', n, rank
                print 'Inven: ', inven
                temp.append(inven)
                n+=1
                for pair in output:
                    print 'Input: ', pair[0], '   Output: ',
pair[1]
    return inventories


def get_L2_affixes(inputs, affixes):
    new_inputs = []
    for inps in inputs:
        temp = []
        for inp in inps:
            temp.append(inp)
            for affix in affixes:
                print inp, affix
                temp.append(affix+inp)
                temp.append(inp+affix)
                temp.append(affix+inp+affix)
        new_inputs.append(temp)
    return new_inputs
```