Verifying Vowel Harmony Typologies

Sara Finley
University of Rochester
Rochester, NY, USA
sfinley@bcs.rochester.edu

Abstract

This paper applies finite state technologies to verify the typological validity of Turbid Spreading, a theory of vowel harmony in Optimality Theory (OT) (Prince & Smolensky, 1993/2004). Previous analyses of vowel harmony in OT have been prone to typological inconsistencies, predicting grammars that do not occur in natural language (Wilson, 2003). However, attempts to eliminate typological pathologies relying on hand-made inputs and candidate sets have been shown to be highly prone to error (Wilson, 2005). Using a modified version of the Contenders Algorithm (Riggle, 2004b), we verify that Turbid Spreading makes typologically valid predictions about the types of harmony processes that may appear in natural language. This modification of the Contenders Algorithm to include complex spreading interactions and intermediate representations demonstrates the utility of computational methods for verifying the typological predictions of complex phonological theories.

1 Introduction

The goal of Optimality Theory (OT) (Prince & Smolensky, 1993/2004) is to understand and explain the mechanisms responsible for linguistic processes. Because it is possible to use constraint rankings to generate a set of possible grammars, OT is fundamentally a theory of cross-linguistic typology.

One of the theoretical assumptions of OT is that it is the job of the grammar to determine which languages are possible and which are not. While the full typology of identifiable languages can never be verified, it is generally agreed that there is a difference between unattested languages that are accidental gaps and unattested languages that are pathological. While both are unattested, accidental gaps are theoretically possible, and might be found given enough time. Pathological languages are languages that are logically possible, but violate general principles of language, and no natural language is expected to contain such pathologies.

Thus, OT assumes that a valid grammar is typologically sound if it does not generate pathologically unattested languages. However, it is extremely difficult to assess the typological validity of phonological analyses because the output of a typological prediction is dependent upon the set of constraints, the output candidates considered, and the underlying forms (inputs) of interest. The theorist must therefore be able to consider all possible inputs, to select an inclusive set of candidates, and to be sure to include the relevant constraints. If any one of these factors is not carefully constructed, the theorist may miss an important typological prediction made by the OT grammar.

These challenges can be significantly diminished through computational tools, such as finite-state techniques. With such tools, it is possible to understand typological predictions that would have likely gone unnoticed without a computational model. This paper presents the results of computational models used to revise and formulate a phonological theory. In this paper, we make use of finite-state methods (specifically Riggle's (2004b) Contenders Algorithm) to understand and verify the typological predictions of a particular theory of vowel harmony, Turbid Spreading (Finley, 2008, in press). Without the computational tools presented in this paper, many unwanted predictions would have been made.

Vowel harmony is a phonological process whereby a particular phonological feature is shared by all vowels in a given lexical domain. For example, in Turkish vowel harmony, if the first vowel of the word is [+Back], all following vowels are [+Back], creating a spreading process whereby [+Back] is spread from the left edge of the word to the right.

Proceedings of the 11th Meeting of the ACL-SIGMORPHON, ACL 2010, pages 9–18,
Uppsala, Sweden, 15 July 2010.

In simple[1] cases of harmony, all possible vo w-els undergo spreading. However, harmony fails when a vowel is unable to take on the spreading feature (e.g., if the language does not allow [–High] vowels to be [+Back]). In these cases the non-participating vowel can either block harmony and create a new spreading domain (as an opaque vowel; [+ – –]) or allow the spreading domain to skip the nonparticipating vowel (as a transparent vowel; [+ – +]).

There are two reasons that vowel harmony in OT is an ideal candidate for the present case study of the use of finite-state techniques to verify linguistic typologies. First, vowel harmony is cross-linguistically widespread, with a clear typology of patterns that are both frequent as well as those that are unattested. Further, the ways in which vowel harmony interacts with other processes (e.g., epenthesis and deletion) are well understood, such that it is possible to differentiate between accidental gaps and pathological unattested languages. For example, direction of spreading in vowel harmony is determined by the featural, morphological or (left or right) edge status of the potential harmony source trigger vowel. There are no languages that determine the direction of spreading by fewest changes from the input to output (referred to as 'majority rules') (Finley & Badecker, 2008). However, this pattern is easy to produce using Agree constraints which merely require adjacent vowels to agree, and do not specify direction. It is these kinds of unattested interactions that an ideal model of vowel harmony in OT should avoid. This ideal model must also be able to account for the major harmony patterns (transparency, opacity, etc.).

Establishing a theory of vowel harmony in OT with both of these properties has been particularly problematic. In addition to 'majority rules' patterns, interactions between non-participating vowels and directionality of spreading have posed a particular challenge. For example, traditional constraints used for vowel harmony (e.g., Align and Agree) predict harmony interactions that do not exist, such as failure to spread to regularly undergoing vowels in the presence of a non-participating vowel, or deletion of a non-participating vowel in order to preserve agreement of vowel features (Wilson, 2003). These harmony pathologies pose a great challenge for vowel harmony and OT in general. Turbid Spreading is a representational approach to vowel harmony in OT that has been designed with this challenge in mind.

The second reason that vowel harmony is an ideal method for studying the interaction of theoretical and computational methods is that vowel harmony requires rich representations. These rich representations pose a unique opportunity to integrate theoretical and computational methodologies. Specifically, we capture these rich representations through the Contenders Algorithm (Riggle, 2004b).

Further, vowel harmony is an important area of research in computational phonology (Bird & Ellison, 1994; Ellison, 1992; Goldsmith & Xanthos, 2009) because the representation of agreement between vowels across consonants poses unique challenges to the learner. This paper differs from previous computational models of vowel harmony because the present work is an instantiation of a generative OT model. The present work focuses on framing work done in theoretical linguistics in a computational framework.

The paper begins with a brief overview of the Contenders Algorithm (Section 2). This is followed by a description of Turbid Spreading and its formalization in finite-state representations (Section 3). Section 4 presents the results of the typological analysis.

2 The Contenders Algorithm

Riggle's (2004a, 2004b) algorithm uses finite-state techniques to find the set of candidates for a given input that are possible optimal outputs under any possible ranking. In order to compute constraint violations, both GEN and the constraints in CON are represented in terms of a finite state transducer. The use of finite representations of infinite sets of strings has important consequences for Optimality Theory. As long as GEN can be represented in terms of finite-state transducers, it is possible to represent the infinite candidate set in terms of a single computation. When all constraints are combined and a single input is evaluated, there will only be a finite set of contenders[2].

The Contenders Algorithm creates a single finite state transducer via the intersection of finite-state transducers for GEN and CON. This combined transducer is an unranked grammar. Be-

[1] Like most phonological processes, vowel harmony is subject to exceptions (Finley, 2010). Future work will incorporate exceptions into computational methods.

[2] See Riggle (2004b) for proof that the list of contenders will always be a finite set.

cause the goal of the algorithm is to produce a list of candidates that could 'win' under some ranking, the algorithm must entertain all possible rankings.

Violations of constraints are instantiated through costs for specific paths in the transducer (e.g., a path that changes a [+F] vowel to a [–F] vowel may have a cost of 1, incurring a single violation). The combined transducer makes it possible to find the constraint profile for any input-output mappings created by GEN. This is the cost of traversing the transducer from start to finish for a given input-output pair. The Contenders Algorithm compares violation profiles for given constraints and candidates, making it possible to predict which violation profiles (candidates) are able to win under some ranking (i.e., which candidates are contenders).

The Contenders Algorithm uses Ellison's (1994) model of finite-state transducers in OT to find the least costly paths through the finite-state grammar. Because each arc of the transducer corresponds to a segment in the string (along with the input-output mapping for that segment), the costs associated with that segment (i.e. constraint violations) are found in each arc. These costs are stored as n-tuples that can be used to compare the costs associated with different candidates.

Riggle's model is based on Dijktra's (1959) shortest path algorithm. Every time a candidate violates a constraint, it increases the 'distance' through the transducer. According to Dijkstra's model, the shortest path through a transducer is also the shortest path through each intermediate step (as each intermediate step serves as a subset of the shortest path). This means that candidates that incur many violations will have the most costly paths. By comparing each candidate's cost for each constraint, it is possible to find which candidates are harmonically bound (i.e. cannot win under any ranking) and which are not (the contenders).

The Contenders Algorithm selects each node of the intersected finite-state transducer and records the cost of each arc outside that node. The cost of visiting that particular node is recorded if that cost is less than or equal to previously recorded costs. After all nodes have been evaluated, a list of the costs associated with each node is produced. The Contenders Algorithm then generates a list of all candidates whose costs do not exceed that of the least-cost list; these are the contenders for a given input.

The output of the Contenders Algorithm for a large set of inputs can be used to create a typological analysis (Bane & Riggle, in press). This typological analysis provides information about the relationship between the different contenders and the rankings that produce them. The typology is formed by inputing the list of contenders for a range of inputs into an algorithm that computes Elementary Ranking Conditions (ERCs) (Prince, 2002). ERCs produce a set of possible ranking interactions from a set of candidates and their violation profiles.

The present paper modifies Riggle's (2004b) model in several ways. First, Riggle's model is relatively simple in terms of the types of segments used. Riggle is able to model epenthesis and deletion of segments listed as /a/ and /b/. In the present model, we include binary vowel features ([±F]) that are active in vowel harmony (in addition to consonants). Second, the number and complexity of constraints are increased. The present model allows for deletion and insertion, as well as feature agreement and featural markedness. Third, the present model adds an intermediate level of representation between the input and the output, thereby increasing the level of complexity in both the constraints and the evaluation. This demonstrates that Riggle's model is capable of handling rich representations, complex phonological processes and multiple assumptions about the architecture of representations in phonology. Thus, the Contenders Algorithm is important for a wide range of problems in phonological theory, and has the ability to bring computational approaches to problems that affect researchers in phonology beyond the computational linguist.

3 Turbid Spreading

Turbid Spreading is a representational approach to vowel harmony based loosely on Turbidity Theory (Goldrick, 2001), a model for opaque representations in phonology. This model uses hidden representations as a method for accounting for incremental phonological processes in a parallel fashion. Turbid Spreading extends this idea of hidden representations. In Turbid Spreading, featural representations for segments have three levels: the underlying representation (UR), the projection level (PR), and the surface level (SR). Relations between the underlying form and the surface form are achieved at the projection level. All segments have a feature value at the projection level. Because we are concerned with spreading between vowels, we focus solely on the representations for vowels. In the present implementation, there is a single harmonic feature of interest (±F) and a secondary unary feature (B) that is penalized when a segment that possess both B and +F feature values (and can therefore block harmony, such as a [+High] vowel that cannot be [+Back]). All ± notations refer to the feature F. Thus, +B is a [+F, B] vowel and –B is a [–F, B] vowel.

All projection level representations are marked ±U, ±R, ±L, and ±P. The ± refers to the feature value for F (+F/–F), and the letter (U, L, etc.) refers to the source of the projection, described below.

The source of the projected feature value can be the underlying form (a faithful representation, marked as ±U, in which +U refers to a +F vowel projected by its underlying form, and –U refers to a –F vowel projected by its underlying form), a neighboring vowel (via spreading, marked as ±L/±R, in which +L refers to a +F vowel projected by the vowel to its left) or the phonetic representation, via the surface level (marked as ±P, in which +P refers to a +F vowel projected by its surface form). Each vowel has one and only one source for its projection value.

In Turbid Spreading, vowel harmony is achieved when the feature value at the projection level of the triggering vowel spreads to an adjacent vowel. In the example of spreading given in Figure 1, the pictoral representation of spreading is given on the left, with the notational features given on the right. The first vowel spreads [+F] to the second vowel, causing the second vowel to be represented as +L at the projection level (because it receives a [+F] feature from the vowel to the left). The underlying form and the surface form do not change as a result of spreading.

[+F] [–F] UR: +F –F
↓
[+F] → [+F] PR: +U +L

[+F] [+F] SR: +F +F

Figure 1: Spreading

An important restriction on spreading is that the features at both sides of the 'arrows' must match (e.g., [–F] → [+F] is prohibited[3]). In other words, vowels may only spread the feature value at the projection level. However, this does not preclude feature values from changing at different levels (e.g., from [+F] in the UR to [–F] in the PR). Allowing changes to feature values at different levels captures both direct spreading processes, as well as opaque interactions between spreading and the surface form. For example, a transparent vowel is created when [+F] spreads to a non-participating vowel (giving +L/+R at the PR) but the non-participating vowel pronounces [–F] at the SR. In this case, the feature values at the projection and pronunciation levels will not match. Transparent vowels therefore satisfy spreading constraints, but violate the constraints requiring the feature values of the surface form and the projection level to match.

For the purpose of formalizing Turbid Spreading into regular expressions for the Contenders Algorithm, we treat each level of representation (UR, PR and SR) as an element of a triple. There are four feature values that appear in the UR and the SR: /+F, –F, +B, –B/. The feature B (potential harmony blocker) is a placeholder for a feature that may or may not spread that harmonic feature. This allows us to place restrictions on which vowels can undergo spreading (e.g., a restriction that non-high vowels cannot undergo back harmony). This secondary feature is important for evaluating the typology of interactions between participating and non-participating vowels.

The projection level (PR) representation contains both featural information as well as the source of spreading. The feature values for F are shortened to be simply +/–. For example, a +F vowel projected by U is written as +U rather than +FU. Thus, ±U implies a faithful ±F feature representation of the underlying form (e.g., –U implies a faithful representation of the –F feature value in the underlying form). ±P implies that the phonetics has caused a change in the representation. ±R implies leftward spreading (the vowel to right spread to the current vowel). ±L implies rightward spreading (the vowel to the left spread to the current vowel). The representation for each string of vowels is written as [UR: PR: SR]. The pictorial representation in Figure 1 is therefore [+F –F: +U +L: +F +F].

We implemented Turbid Spreading in the Contenders Algorithm by using finite state implementations of GEN and of the constraints known to interact with spreading[4]. Each arc of the transducer represents a single segment, pre-

[3] The present model does not account for dissimilation, but we assume that will be accounted for by some other mechanism, and is subject to future work.

[4] Text versions of the FSA's can be found at: http://www.cog.jhu.edu/grad-students/finley/fsa.pdf

sented as a tuple. A '.' notation indicated that the position in the tuple could be filled by any feature value. Non-crucial arcs were removed from the diagrams of finite-state transducers (but were included in the formal analysis). These include the potential for vowel epenthesis (notated as [-]) and vowel deletion (notated as an [x]). Note that the symbols 'x' and '-' are used solely for 'bookkeeping' purposes in the FSA's and are not necessarily part of the phonological representation.

We also removed several arcs allowing for the presence of consonants (represented as [C]). Because projection from the surface form (+P/–P) works the same as projection from the UR (in terms of vowel harmony), these are left out of the descriptions (but were included in the formal analysis).

The transducer for Gen is given in Figure 2. This finite state transducer accepts strings of concatenated vowels for all potential inputs. This transducer provides the basis for restrictions on the representations for spreading. One such restriction is that the feature value of the projection must match the source feature value. For example, a vowel with a [+U] projection must have [+F] in the UR (e.g., arc 0 to 1). A second restriction is a practical one; the first vowel in a word cannot be projected by the vowel to its left (because such a vowel does not exist). The third restriction is that vowels have one and only one projection. Gen only produces segments that have a single value at the projection.

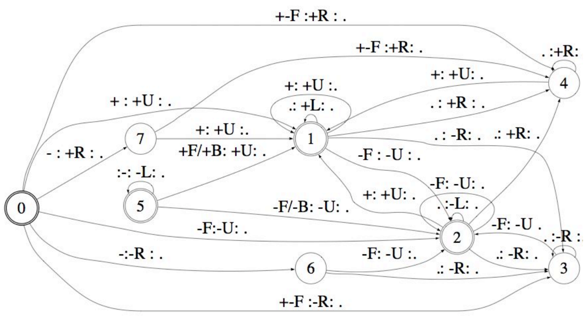

Figure 2: GEN

Spreading is initiated by a vowel whose projection is its underlying form (+U/–U). A vowel may only be projected by +L if it follows a vowel that is projected by +U (and likewise for –L). This ensures that the only initiator for rightwards spreading is a vowel projected by its underlying form.

In Turbid Spreading, deletion of a segment entails deletion of only the surface form; all vowels with a UR have a representation at the PR. Epenthesis can occur at either the PR level (requiring representations at both the PR and SR) or the SR (requiring only a representation at the SR). Epenthetic vowels at the PR level undergo spreading (e.g., arc 0 to 6), whereas epenthetic vowels at the SR only are transparent to spreading. The difference between epenthetic and deleted vowels is based on the fact that epenthetic vowels may interact with spreading (at the projection level) or be transparent (and appear only at the pronunciation level), but deleted vowels may not interact with spreading (and therefore appear only at the pronunciation level).

Rightwards spreading is instantiated in the arcs from state 0 to states 3, 4, 6 and 7. Transitions from 0 to 6 and 0 to 7 involve epenthetic vowels (marked with a /-/ in the UR). Arcs from 0 to 3 and 0 to 6 involve spreading -F to the left (/-R/ at the PR). Arcs from 0 to 7 and 0 to 4 involve spreading +F to the right (/+R/ at the PR). In order for a /+R/-R/ projected vowel to reach a final state, the source of spreading must be +U/-U, which is reflected in the arcs from 3 to 2, and 4 to 1. Spreading from the left to right involves a vowel projected by its underlying form (transitions 0 to 1 and 0 to 2). From there a vowel may be projected as +L (state 1) or as –L (state 2).

Constraints Turbid Spreading is instantiated through several constraints. SPREAD-R and SPREAD-L initiate vowel harmony. ID[F]-UR regulates featural identity between the underlying representation and the projection. ID[F]-SR regulates featural identity between the surface representation and the projection. Constraints that govern epenthesis and deletion are MAX, DEP, and *CC. The finite state transducers of these constraints assign violation marks (e.g., -1) to the output whenever a violation of the constraint is encountered.

ID[F]-UR is violated once for every vowel not projected by its underlying form. Any vowel that has an underlying form (i.e., not an epenthetic vowel), but is not marked with ±U at the PR incurs a violation.

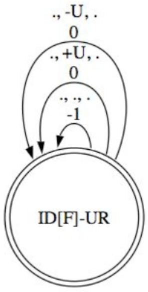

Figure 3: ID[F]-UR

*B is a placeholder for a featural markedness constraint (e.g., *[+Back, –High]). This constraint is violated when a vowel is marked ±B in the input and is +F in the output (e.g., a [–High] vowel becoming [+Back]). All other representations do not receive a violation[5].

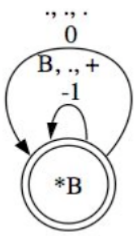

Figure 4: *B

The ID[F]-SR constraint is violated whenever the feature values at the projection and the pronunciation level do not match. For example, if the pronunciation is [+], the projection must be +U, +L, +R or +P.

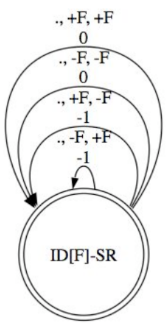

Figure 5: ID[F]-SR

The ‘x’ symbol is used to denote deleted vowels, which violate MAX, which assigns a violation if ‘x’ appears in the pronunciation.

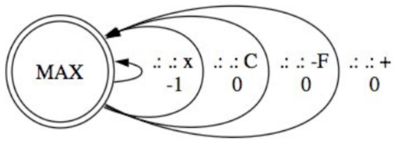

Figure 6: MAX

DEP is the constraint violated by epenthesis, represented by the symbol ‘-‘ in the underlying form. DEP searches for any vowel with (-) in the UR and assigns a violation for each feature value that appears on the projection and pronunciation levels. Epenthesis at the pronunciation level incurs two violations of DEP, but epenthesis at the projection level incurs one violation.

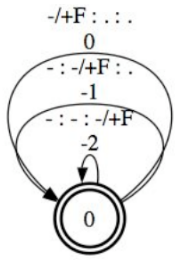

Figure 7: DEP

I assume that epenthesis is driven by the markedness constraint *CC[6]. This constraint scans the pronunciation level for two consonants in a row, and assigns a violation for every pair of consonants. *CC requires two states because *CC is violated only when there are two consecutive consonants, making one state for the first consonant (no violations), and a second state for an adjacent consonant (a violation).

[5] This constraint assumes that no vowels may lose their /B/ specification from the input to the output (e.g., change from [–HIGH] to [+HIGH] in order to allow spreading of [+Back]). This process is called ‘re-pairing’ (Bakovic, 2000), and is subject to future research.

[6] In addition to *CC, other constraints such as *#C or *C# may trigger epenthesis. For simplicity, these additional constraints are not included in the present analysis.

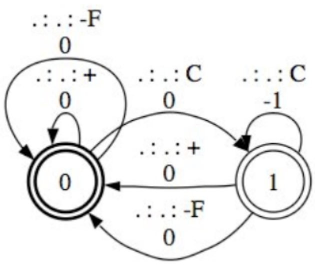

Figure 8: *CC

Violations for both spreading constraints are assigned directionally such that a violation on the first vowel is more severe than violations later in the word (Eisner, 2000), formalized in a simplified version where violations at different parts of the word are greater than other parts of the word. In order to prevent 'gang' effects, violations are assigned exponentially such that for a three-vowel input, violation on the second vowel incurs 100 violations, while a violation on the third vowel incurs only 10 violations. This simplified version of directional evaluation only allows for a finite number of vowels in the input. However, because the theory is tested with inputs of 3 and 4 vowels in length, these simplified transducers capture the data analyzed here. Future work will analyze directional spreading for an unlimited number of vowels in the input.

SPREAD-R is satisfied if a vowel projects an L (+/–L only occurs if a vowel spreads rightwards). Spreading is represented in terms of the target of spreading (e.g., a [+L] vowel is the target of spreading). Because initial vowels cannot be a target of rightward spreading (as there is no vowel to the left), the initial vowel automatically satisfies SPREAD-R.

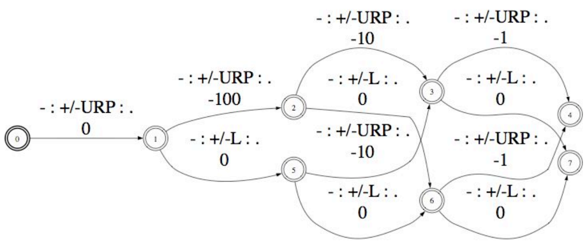

Figure 9: SPREAD-R

From state 1, vowels that project an L satisfy SPREAD and move to state 5. All other vowels move to state 2 and incur 100 violations. From states 2 and 5, if the third vowel satisfies SPREAD, it moves to state 6. If the final vowel satisfies SPREAD, it moves to state 7. If the third vowel fails to satisfy SPREAD, it moves to state 3 (from state 5 or 2) and incurs 10 violations. If the fourth vowel fails to satisfy SPREAD, it moves to state 4 (from states 6 or 3) and incurs 1 violation.

One might assume that SPREAD-L simply is a reversed version of SPREAD-R. However, this simple reversal is not possible because the opposite vowels trigger spreading for each constraint. In SPREAD-R, the initial vowel is the optimal trigger for harmony, but for SPREAD-L the final vowel is the optimal trigger for harmony.

The final vowel always satisfies SPREAD-L because final vowels cannot be targets for leftward spreading. Thus, the final vowel (state 7) never incurs a violation. If the first vowel is projected by +R or -R, then it satisfies SPREAD-L (state 3), otherwise it violates the constraint, and incurs 1 violation (state 4). If the second non-final vowel violates SPREAD-L, it moves to state 2 and incurs 10 violations. If the third non-final vowel violates SPREAD-L, it moves to state 3 and incurs 100 violations.

Violations of harmony from epenthetic vowels are assigned based on the position of the word. If an epenthetic vowel does not get its projected feature from the right, it will incur a violation of SPREAD-L. Epenthesis before the initial vowel incurs 1 violation, epenthesis after the initial vowel incurs 10 violations, etc.

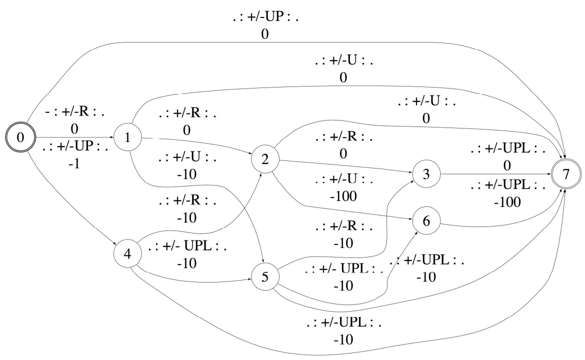

Figure 9: SPREAD-L

4 Results

The finite state transducers implementing GEN and the constraints were fed into the Contenders Algorithm. This was a modified version of Riggle's java script program[7]. This program computes the contenders for a single input over the grammar. While the finite-state transducers represent an infinite candidate set, it would be impossible to compute contenders for every possible input. We limited the input to 4 vowels be-

[7] Thanks to Colin Wilson for these modifications.

cause all previously reported pathologies did not change for words longer than four vowels (3 with epenthesis)[8]. We used Microsoft Excel to co m-pute all possible feature combinations for up to four vowels (+F, –F, +B, –B) without epenthesis. There were 256 combinations with 4 vowels in the input, 64 combinations with 3 vowels, 16 combinations with 2 vowels, and 4 with 1 vowel in the input. The input list with epenthesis used the vowel combinations for up to 3 vowels, and CC clusters were inserted at the left edge, right edge, and medially (when applicable).

The results of the Contenders Algorithm were fed into the Erculator program for computing typologies using Elementary Ranking Conditions (ERCs) (Riggle, 2007). Without epenthesis, there was a typology containing 16 languages: 6 spread right, 6 spread left and 4 with no spreading. For the no spread cases, there was one language that allowed the marked segment ([+B]), and three that did not. In the three that did not allow [+B] in the output, underlyingly /+B/ segments were treated differently. In one language, underlyingly /+B/ segments got their [–F] feature from the vowel to its left, in another language, underlyingly /+B/ segments got their [–F] feature from the vowel to its right, and in the third language, underlyingly /+B/ segments got their [–F] feature from the pronunciation level.

The six spread-right and spread-left languages were identical except for direction of spreading. In one language all vowels participated in harmony. The second language was a case of 'allophonic harmony'. In this case, a [+B] vowel only appears as a result of harmony. That is, harmony creates allophones of a phoneme that would otherwise not appear on the surface. Non-participating vowels were transparent in the third and fourth cases. In the third case, an underlyingly +B vowel was changed via spreading; in the fourth case, an underlyingly +B vowel was changed at the pronunciation level. In the fifth and sixths cases, non-participating vowels were opaque, blocking harmony and starting a new harmonic domain. In the fifth case, underlyingly /+B/ segments changed to [–F] via projection from the surface. In the sixth language, underlyingly /+B/ segments could undergo spreading of [–F] from either the left or the right, if possible.

With epenthesis, the predicted typology contained 68 languages. There were 16 languages with no epenthesis, 16 languages with epenthesis always on the projection level and 16 languages with epenthesis at the pronunciation level (giving 48 languages). Each of these sets of 16 languages corresponded to the 16 languages with no epenthesis above. The final 20 languages were from cases in which epenthesis occurred at the projection level only if spreading were possible from the vowel to the left (10 languages) or the vowel to the right (10 languages). These 20 languages differed depending on how non-derived vowels behaved. There were two sets of 4 no-spread languages, 6 spread-right languages, and 6 spread-left, described above.

Pattern	**Examples**
1. All vowels participate	Kalenjin (Local & Lodge, 1996) Degema (Elugbe, 1984)
2. Transparent Vowels	Hungarian (Goldsmith, 1985) Finnish (Goldsmith, 1985)
3. Opaque Vowels	Mongolian (Goldsmith, 1985) Turkish (Underhill, 1976)
4. Bi-Directional Harmony	Lango (Woock & Nooonan, 1979) Kalenjin (Local & Lodge, 1996) Turkana (Dimmendaal, 1983)
5. Allophonic Harmony	Pasiego (Penny, 1969) Akan (Clements, 1981) Kinande (Archangeli & Pulleyblank, 2002) Nawuri (Casali, 2003)
Epenthetic Vowels	
6. Transparent	Karchevan (Vaux, 1995) Agulus (Vaux, 1998)
7. Undergo Harmony	Turkish (Clements & Sezer, 1982; Underhill, 1976) Yawelmani (Archangeli, 1988) Yoruba (Archangeli & Pulleyblank, 1989)
8. Directional Harmony	Levantine Arabic (Kenstowicz, 1981) Mohawk (Postal, 1968), Sesotho (Rose & Demuth, 2006)
9. Epenthetic Vowels Only	Ponapean (Kitto & DeLacy, 1999) Barra Gaelic (Sagey, 1987) Marash (Vaux, 1998)

Table 1. Patterns of harmony languages

Importantly, all 68 of these languages represent possible or known languages; none of these languages share the properties of pathological typological predictions described by Wilson

[8] Pilot tests with longer inputs did not change the results.

(2003). An important result of these computations is that epenthesis is never blocked by a failure to participate in harmony, a prediction that previous analyses of vowel harmony incorrectly predicted to be possible (Wilson, 2003).

Because the resulting languages varied systematically, we were able to divide the 68 languages into nine different patterns. Examples from real languages are presented in Table 1. The first pattern is that all vowels participate in harmony; there are no non-participating vowels. The second and third patterns are vowel harmony languages with nonparticipating vowels, either transparent to harmony (case 2) or opaque to harmony (case 3). Case 4 occurs when spreading applies both from right-to-left as well as left-to-right. Case 5 involves allophonic harmony, discussed above. Cases 6-9 apply to epenthetic vowels. In case 6, harmony skips epenthetic vowels. In case 7, harmony applies to epenthetic vowels as if they were an underlying vowel. In case 8, harmony applies to epenthetic vowels, but directionally (e.g., the epenthetic vowel gets its features from the right or left, or defaults if there is no vowel to spread to the epenthetic vowel). Case 9 occurs when harmony does not apply to underlying vowels in the language, and only epenthetic vowels undergo harmony.

The important result found in these 9 case patterns is that all the major harmony phenomena are predicted (directionality, epenthesis, transparency and opacity), without predicting typologically implausible languages. This is an important result because many previous theories of vowel harmony in OT made pathological predictions when harmony interacted with non-participating vowels, deletion and epenthesis. For example, alignment constraints predict failure to epenthesize a vowel in the presence of a non-participating vowel (Wilson, 2003). Such pathologies are not found in Turbid Spreading.

While there other instances of vowel harmony that are not covered in the present analysis, the present approach provides a mechanism for understanding the typology of vowel harmony processes and the mechanisms that produce the attested and the unattested patterns.

It is important to note that while the present results are successful, the model is a result of revisions based on previous iterations of the Contenders Algorithm. Many of the unwanted predictions in previous models could not have been found without the use of the computational tools used in this paper.

5 Conclusion

Computations over finite-state transducers made it possible to compute a complete typology of vowel harmony interactions, including interactions of vowel harmony and epenthesis. The computational model verified that Turbid Spreading only predicts languages known to be attested in natural language, but does predict the common pathologies known to be problematic for previous vowel harmony analyses in OT.

Because we used all possible vowel combinations for up to four vowels, we can be fairly certain that all relevant inputs were considered. Because the Contenders Algorithm models the infinite candidate set in GEN, we can be certain that the relevant candidates were considered. This paper demonstrates the power of computational tools for measuring and evaluating theories of phonological phenomena.

Acknowledgments

I am indebted to Colin Wilson, who provided crucial guidance and encouragement. I am also grateful to Jason Riggle, who laid the groundwork for this project. In addition, I am extremely grateful to Paul Smolensky for his help implementing the inputs using Excel, and his guidance in forming Turbid Spreading. I am also thankful to Matt Goldrick, Joan Chen-Main, Gaja Jarosz-Snover, Rebecca Morley, and Neil Bardhan for their helpful comments. Funding was provided by: NSF IGERT and NIH Grant DC00167. All errors are my own.

References

Archangeli, D. (1988). *Underspecification in Yawelmani Phonology and Morphology*. New York & London: Garland Publishing, Inc.

Archangeli, D., & Pulleyblank, D. (1989). Yoruba vowel harmony. *LI, 20*, 173-217.

Archangeli, D., & Pulleyblank, D. (2002). Kinande vowel harmony: domains, grounded conditions and one-sided alignment. *Phonology, 19*, 139-188.

Bakovic, E. (2000). *Harmony, dominance and control.* Unpublished doctoral dissertation, Rutgers University.

Bane, M., & Riggle, J. (in press). The typological consequences of weighted constraints. *Proceedings of the 45th Annual Meeting of the Chicago Linguistics Society*.

Bird, S., & Ellison, T. M. (1994). One-level phonology: Autosegmental representations and rules as finite automata. *Computational Linguistics, 20*(1), 55-90.

Casali, R. F. (2003). [ATR] value asymmetries and underlying vowel inventory structure in Niger-Congo and Nilo-Sahaan. *Linguistic Typology, 7*, 307-382.
Clements, G. N. (1981). Akan vowel harmony: A non-linear analysis. *Harvard Studies in Phonology, 2*, 108-177.
Clements, G. N., & Sezer, E. (1982). Vowel and consonant disharmony in Turkish. In v. d. H. a. Smith (Ed.), *The structure of Phonological Representations* (Vol. II, pp. 213-255). Dordrecht: Foris.
Dijkstra, E. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik, 1*, 269-271.
Dimmendaal, G. J. (1983). *The Turkana language*. Dordecht: Foris.
Eisner, J. (2000). Directional constraint evaluation in Optimality Theory. *Proceedings of the 18th International Conference on Computational Linguistics (COLING)*, 257-263.
Ellison, M. T. (1992). *Learning vowel harmony.* Paper presented at the SHOE Workshop, ITK, Tilburg, Netherlands.
Ellison, M. T. (1994). Phonological derivation in Optimality Theory. *Proceedings of the 15th International Conference on Computational Linguistics*, 1007-1013.
Elugbe, B. O. (1984). Morphology of the gerund in Degema and its reconstruction in proto-Edoid. *Studies in African Linguistics, 15*(1), 77-89.
Finley, S. (2008). *Formal and cognitive restrictions on vowel harmony.* Unpublished doctoral dissertation, Johns Hopkins University.
Finley, S. (2010). Exceptions in vowel harmony are local. *Lingua, 120*, 1549-1566.
Finley, S. (in press). The interaction of epenthesis and vowel harmony. *Proceedings of the 44th Annual Meeting of the Chicago Linguistics Society*.
Finley, S., & Badecker, W. (2008). Analytic biases for vowel harmony languages. *WCCFL, 27*, 168-176.
Goldrick, M. (2001). Turbid output representations and the unity of opacity. *NELS, 30*, 231-245.
Goldsmith, J. (1985). Vowel harmony in Khalka Mongolian, Yaka, Finnish and Hungarian. *Phonology Yearbook, 2*, 253-275.
Goldsmith, J., & Xanthos, A. (2009). Learning phonological categories. *Language, 85*(1), 4-38.
Kenstowicz, M. (1981). Vowel harmony in Palesitinian Arabic: A suprasegmental analysis. *Linguistics, 19*, 449-465.
Kitto, C., & DeLacy, P. (1999). Correspondence and epenthetic quality. In C. Kitto & C. Smallwood (Eds.), *Proceedings of AFLA VI*. Holland: Academic Graphics.
Local, J., & Lodge, K. (1996). Another travesty of respresentation: Phonological representation and phonetic interpretation of ATR harmony in Kalenjin. *York Papers in Linguistics, 17*, 77-117.
Penny, R. J. (1969). Vowel harmony in the speech of Montanes de Pas. *ORBIS- Bulletin International de Documentation Linguistique*, 148-166.
Postal, P. M. (1968). Mohawk vowel doubling. *International Journal of American Linguistics, 35*, 291-298.
Prince, A. (2002). Arguing optimality. In A. Coetzee, A. Carpenter & P. De Lacy (Eds.), *Papers in Optimality Theory II*. Amherst, MA: GLSA.
Prince, A., & Smolensky, P. (1993/2004). *Optimality theory: Constraint interaction in generative grammar*. Cambridge: Blackwell.
Riggle, J. (2004a). Contenders and learning. In B. Schmieser, A. K. Chand & A. Rodriquez (Eds.), *WCCFL* (Vol. 23, pp. 101-114). Somerville, MA: Cascadilla Press.
Riggle, J. (2004b). *Generation, recognition and learning in finite-state Optimality Theory.* Unpublished doctoral dissertation, UCLA.
Riggle, J. (2007). Erculator: http://clml.uchicago.edu/?page_id=11.
Rose, Y., & Demuth, K. (2006). Vowel epenthesis in loanword adaptation: Representational and phonetic considerations. *Lingua, 116*, 1112-1139.
Sagey, E. (1987). *Non-constituent spreading in Barra Gaelic*.Unpublished manuscript, UC Irvine.
Underhill, R. (1976). *Turkish grammar*. Cambridge: MIT Press.
Vaux, B. (1995). Vowel harmony in the Armenian dialect of Karchevan. *NSL, 9*(1-9).
Vaux, B. (1998). *The phonology of Armenian*. Oxford: Clovendon Press.
Wilson, C. (2003). *Analyzing unbounded spreading with constraints: marks, targets, and derivations*.Unpublished manuscript, UCLA.
Wilson, C. (2005). *Localizing constraint violation: Theoretical consequences of myopic spreading*.Unpublished manuscript, Paper presented at HOWL3, Johns Hopkins.
Woock, E. B., & Nooonan, M. (1979). Vowel harmony in Lango. *CLS, 15*, 20-29.